3-2020

Dynamic Speed and Separation Monitoring with On-Robot Ranging Sensor Arrays for Human and Industrial Robot Collaboration

Shitij P. Kumar
spk4422@rit.edu

Dynamic Speed and Separation Monitoring with On-Robot Ranging Sensor Arrays for Human and Industrial Robot Collaboration

by

Shitij P. Kumar

A dissertation submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in Engineering

Department of Electrical and Microelectronic Engineering
Kate Gleason College of Engineering

Rochester Institute of Technology
Rochester, New York
March 2020

Dynamic Speed and Separation Monitoring with On-Robot Ranging Sensor Arrays for Human and Industrial Robot Collaboration

by

Shitij P. Kumar

Committee Approval:

We, the undersigned committee members, certify that we have advised and/or supervised the candidate on the work described in this dissertation. We further certify that we have reviewed the dissertation manuscript and approve it in partial fulfillment of the requirements of the degree of Doctorate of Philosophy in Engineering.

Dr. Ferat Sahin
Professor, Department of Electrical and Microelectronic Engineering
Advisor

Dr. Sohail Dianat
Dept. Head, Professor, Department of Electrical and Microelectronic Engineering
Committee Member

Dr. Esa Rantanen
Associate Professor, Department of Psychology
Committee Member

Dr. Raymond Ptucha
Assistant Professor, Department of Computer Engineering
Committee Member

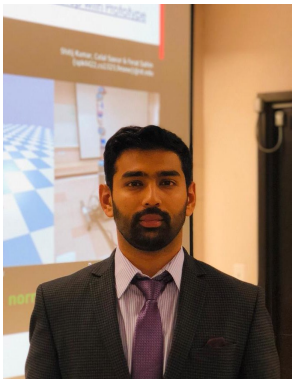
Dr. Donald J. Schertler
Senior Optical Scientist, RPC Photonics
Committee Member

Dr. Ryan Bowen
Chief Technology Officer, NRGXP LLC
Committee Member

Certified by:

Dr. Edward Hensel Jr.
Director, PhD in Engineering program

Biographical Note



Shitij Kumar works primarily in the field of robotics and human-machine interface. His research is about Human-Robot Interaction (HRI) in industry and automation. He works with human-in-the-loop systems, where he programs and studies robot behavior and its effect on human operators sharing and working in the robot workspace. Before this, he has worked on a Brain-Computer Interface(BCI) and studied machine learning techniques and its applications in Assistive Technologies. Shitij's research interests lie in human-robot collaboration, machine learning, computer vision, and embedded system design.

Shitij Kumar received his Bachelors in Technology in the field of Information and Communication Technology (ICT) from DAIICT, Gandhinagar, India. There he majored in digital signal processing and embedded system design. Shitij graduated with a Masters in Science in Electrical Engineering from Rochester Institute of Technology (RIT), Rochester, NY, USA. There he worked as a teaching and research assistant at the Multi-Agent Biorobotics Laboratory.

He went on to pursue his Ph.D. in Engineering at RIT, where he led the development of the first human-robot collaboration (HRC) lab at RIT under the supervision of his advisor Dr. Ferat Sahin. As the Lead Researcher and Manager of the CM Collaborative Robotics Laboratory (CMCRL), he worked with and mentored a team of graduate students on research addressing challenges of human safety, trust-in-automation, level of human-robot interaction and productivity in the field of HRC in industry. He has developed a distributed system to monitor human actions and psychophysiological responses and control multiple robots in the lab during HRC experiments. He has worked as a lead researcher and tester on sponsored projects relating to automation using collaborative robots in industry. He has published his research in leading conferences such as IEEE Systems, Man, and Cybernetics (SMC) and the IEEE Conference on Automation Science and Engineering (CASE).

shitijpkumar@gmail.com kumarshitij.com mabl.rit.edu/cmcr
github.com/spk4422 linkedin.com/in/kumarshitij orcid.org/0000-0002-7210-113X

Abstract

Kate Gleason College of Engineering
Rochester Institute of Technology

Degree: Doctor of Philosophy **Program:** Engineering PhD
Author: Shitij P. Kumar **Advisor:** Dr. Ferat Sahin
Dissertation Title: Dynamic Speed and Separation Monitoring with On-Robot Ranging Sensor Arrays for Human and Industrial Robot Collaboration

This research presents a flexible and dynamic implementation of Speed and Separation Monitoring (SSM) safety measure that optimizes the productivity of a task while ensuring human safety during Human-Robot Collaboration (HRC). Unlike the standard static/fixed demarcated 2D safety zones based on 2D scanning LiDARs, this research presents a dynamic sensor setup that changes the safety zones based on the robot pose and motion. The focus of this research is the implementation of a dynamic SSM safety configuration using Time-of-Flight (ToF) laser-ranging sensor arrays placed around the centers of the links of a robot arm. It investigates the viability of on-robot exteroceptive sensors for implementing SSM as a safety measure. Here the implementation of varying dynamic SSM safety configurations based on approaches of measuring human-robot separation distance and relative speeds using the sensor modalities of ToF sensor arrays, a motion-capture system, and a 2D LiDAR is shown. This study presents a comparative analysis of the dynamic SSM safety configurations in terms of safety, performance, and productivity. A system of systems (cyber-physical system) architecture for conducting and analyzing the HRC experiments was proposed and implemented. The robots, objects, and human operators sharing the workspace are represented virtually as part of the system by using a digital-twin setup. This system was capable of controlling the robot motion, monitoring human physiological response, and tracking the progress of the collaborative task. This research conducted experiments with human subjects performing a task while sharing the robot workspace under the proposed dynamic SSM safety configurations. The experiment results showed a preference for the use of ToF sensors and motion capture rather than the 2D LiDAR currently used in the industry. The human subjects felt safe and comfortable using the proposed dynamic SSM safety configuration with ToF sensor arrays. The results for a standard pick and place task showed up to a 40% increase in productivity in comparison to a 2D LiDAR.

Dedication

I dedicate this work to

My Dad,
for his wisdom, inspiration, and support,
for teaching me to follow my principles and never to stop learning.

My Mom,
for her love, care, and sacrifices she has made for my brother and me.

My amazing wife, Swati,
for her love, support, and encouragement.
Without her, this work would not have completed in time.

My late Grandfather,
my first teacher and fellow adventurer,
who taught me to replace the fear of the unknown with curiosity.

My Grandmother,
for being a loving and caring soul.

My brother, Sangam,
for always having my back.

My professor, Ferat,
for having confidence in me, teaching me how to see things differently,
and for giving me the platform and freedom to learn.

My closest friends,
for their friendship, help, and motivation.

God,
for giving me the strength and good health to pursue my dreams,
and for all your blessings to me, my family, my teachers, and friends.

Acknowledgments

Through my years here, there are many I would like to thank.

Above all, I am grateful for the guidance and generosity that has and continues to be provided by my advisor Dr. Ferat Sahin. His words of inspiration and wisdom have found themselves contributing to the basis of my success.

I would like to thank my dissertation committee, Dr. Dianat, Dr. Rantanen, Dr. Ptucha, Dr. Schertler, and Dr.Bowen, who have each provided helpful feedback and guidance with this research.

I am very grateful to my wife, Swati, who has stood by me through all my travails, my absences, my ups, and downs. She has been incredibly supportive of me throughout this endeavor. I am thankful to my parents and my brother for their continued support and encouragement.

I would like to convey special thanks to the Electrical and Microelectronic Department at RIT for the resources and help provided for my research. I am especially grateful to Ken Snyder, for his candor and resourcefulness, whose support and help made my research a lot easier. Also James Stefano, for his practical approach to things and help with all the computers, networks and software troubles I faced.

I would also like to thank our sponsors Century Mold Inc. for sponsoring the Collaborative Robotics Research Lab (CMCR), RPC Photonics Inc., and NyPro for the grant for this work.

I am grateful to all my peers and fellow research assistants at the Multi-Agent Biorobotics Laboratory. Each of you have contributed your unique roles in the completion of this work.

Celal Savur, thank you for being an excellent colleague and a great friend. The software architecture of this work was only possible because of your guidance, thanks.

Sarthak Arora, thank you for your insights and help with the experiments. I am glad and excited you and Celal would be continuing this research further.

Tuly Hazbar, thank you for the insightful discussions on human-robot collaboration.

Anmol Modur, thank you for helping me set up the CMCR lab, you are the best. Odysseus and Eric, I am grateful for the work we have done in designing and building the Time-of-Flight sensor prototype.

For the rest not mentioned, I thank you all.

List of Contributions

Highlights of the Work

- A direct computation method with low complexity for measuring the human-robot minimum separation distance, and estimating the relative human-robot speeds using on-robot ranging sensor arrays and the robot kinematics, for implementation of Dynamic Speed and Separation Monitoring.
- A Cyber-Physical System (a system-of-systems) capable of controlling and monitoring the robot motion, tracking the human movements, and the progress of the human-robot collaboration (HRC) task during HRC experiments. This system leveraged the digital-twin implementation of the HRC task for analyzing the HRC experiments.
- An octree-based volumetry method for quantifying the on-robot sensor's sensing volume (the volume of the sensor's detection zone) and its coverage of the robot's workspace
- A modular design and implementation of a 32-node ToF sensor array as a minimum viable product for use as an electronic safety device to implement dynamic SSM with collaborative and industrial robots.

Publications

[1] **S. Kumar**, S. Arora, and F. Sahin, **'Speed and separation monitoring using on-robot time-of-flight laser-ranging sensor arrays'**, In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1684–1691, 2019 [1].

[2] **S. Kumar** and F. Sahin, **'Sensing volume coverage of robot workspace using on-robot time-of-flight sensor arrays for safe human robot interaction'**, in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 378–384. IEEE, 2019 [2].

[3] C. Savur*, **S. Kumar***, and F. Sahin, **'A framework for monitoring human physiological response during human robot collaborative task.'**, in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 385–390. IEEE, 2019 [3].

[4] O. Adamides*, A. Modur*, **S. Kumar***, and F. Sahin, **'A time-of-flight on-robot proximity sensing system to achieve human detection for collaborative robots'**, in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1230–1236. IEEE, 2019 [4].

[5] C. Savur*, **S. Kumar***, S. Arora, T. Hazbar, and F. Sahin, **'HRC-SoS: Human robot collaboration experimentation platform as system of systems'**, in *2019 14th Annual Conference System of Systems Engineering (SoSE)*, pages 206–211, May 2019 [5].

[6] **S. Kumar**, C. Savur, and F. Sahin,**'Dynamic awareness of an industrial robotic arm using Time-of-Flight laser-ranging sensors'**, in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2850–2857. IEEE, 2018 [6].

[7] **S. Kumar** and F. Sahin, **'A framework for an adaptive human-robot collaboration approach through perception-based real-time adjustments of robot behavior in industry'**, in *2017 12th System of Systems Engineering Conference (SoSE)*, pages 1–6. IEEE, 2017 [7].

(represents equal contribution)*

Preprints

[8] S. Arora, **S. Kumar**, and F. Sahin, **'Human position detection & tracking with on-robot Time-of-Flight laser ranging sensors'**, (Preprint) in *arXiv preprint arXiv:1909.09750*, 2019 [8].

[9] T. Hazbar, **S. Kumar**, and F. Sahin, **'Cyber-physical testbed for human-robot collaborative task planning and execution'**, (Preprint) in *arXiv preprint arXiv:1905.00199*, 2019 [9].

[10] **S. Kumar**, S. Arora, C. Savur and F. Sahin, **'Dynamic Speed and Separation Monitoring using on-robot ranging sensor arrays'**, *Article in Preprint*

[11] **S. Kumar**, C. Savur, T. Hazbar, S. Arora and F. Sahin, **'A survey of Human-Robot Collaboration research in industry'**, *Article in Preprint*.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, an overview of the importance of human-robot collaboration (HRC) in the industry and its challenges is presented. This research addresses the challenges of ensuring the safety of a human operator during a human-robot collaboration task. The main objective and organization of this dissertation is outlined in this chapter.

1.1 The Significance of Human-Robot Collaboration

Industrial robots working in isolation in a highly automated system are valued for their high productivity. The shortcomings of these pure robotic cells become more apparent when flexibility in production is required to respond to varying production volumes and customized product demands. Complete automation is highly productive, but its costly to set up and difficult to change. On the other hand, manual production, although flexible is slower and prone to human error. Hence in industry, smarter automation methods are required that leverage the dexterity, flexibility, and decision-making capability of a human to speed, precision, and power of a robot. In industry, the need for flexibility in production has resulted in the acceptance of Human-Robot Collaboration as a viable alternative (represented in Figure 1.1).

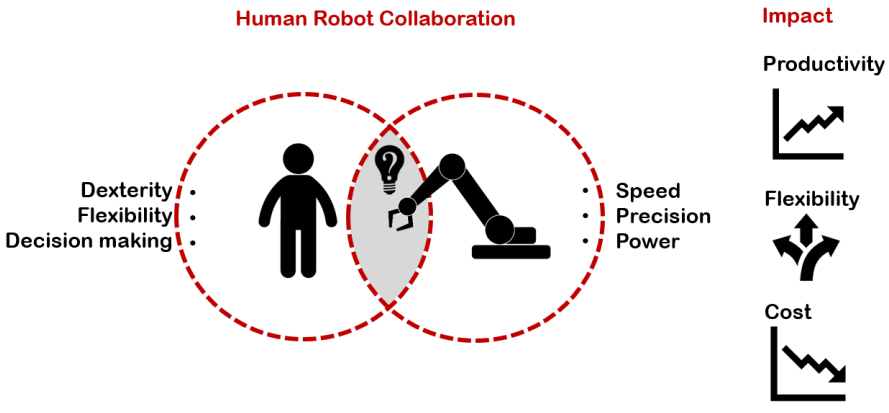


Figure 1.1: Significance of Human-Robot Collaboration and its positive impact in industry.

Next, we explore the economic and research motivations of HRC and the proposed research.

1.1.1 Economic Motivation

The changing societal and marketing trends are some of the main propellants to the acceptance of the idea of human-robot collaboration in industry. Currently, industries would like to attract customers by having individuality and differentiation in their production process with respect to their competitors and peers. Having robots as part of the production has always been a symbol of technological aptitude and ability of the industry, thereby being attractive to the customers.

In response to volatile markets and ever-increasing high mix/low volume production demands, it has necessitated flexibility in production and manufacturing in industry. The resulting market trends have required industries to move away from the concepts of 'mass-production' towards the concept of 'mass-customization' [32, 33]. The current market demands have also increased the number of product variants and product upgrades resulting in a decrease of the product lifetime. This poses a lot of challenges to the industrial production process. There is a need for efficient handling of large variants in the production and a short lifetime of a product.

Hence, a human-robot collaborative production has more potential to handle and balance the current challenges of industrial production. Figure 1.2(a) shows the current market trend by showing the relation between the number of units of a product model produced to that of the product life span [1]. A chart representing the balance of productivity, lot size[2], the flexibility of automation and variants such as the complexity of automation during manual, human-robot hybrid, and automatic assembly is shown in Figure 1.2(b).

According to a report [34] and the Bureau of Labor Statistics [35], the manufacturing industry will have a shortage of skilled labor because of the retirement of workers due to age, the inability to keep up with the changing technology, and the negative public perception of manufacturing jobs. Hence, the manufacturing industry is changing to incorporate

[1] Figures and charts are adapted from the study performed by ABB to explain the economic impacts of human-robot collaboration in industry [10].

[2] 'Lot size' refers to the quantity of an item ordered for delivery on a specific date or manufactured in a single production run.

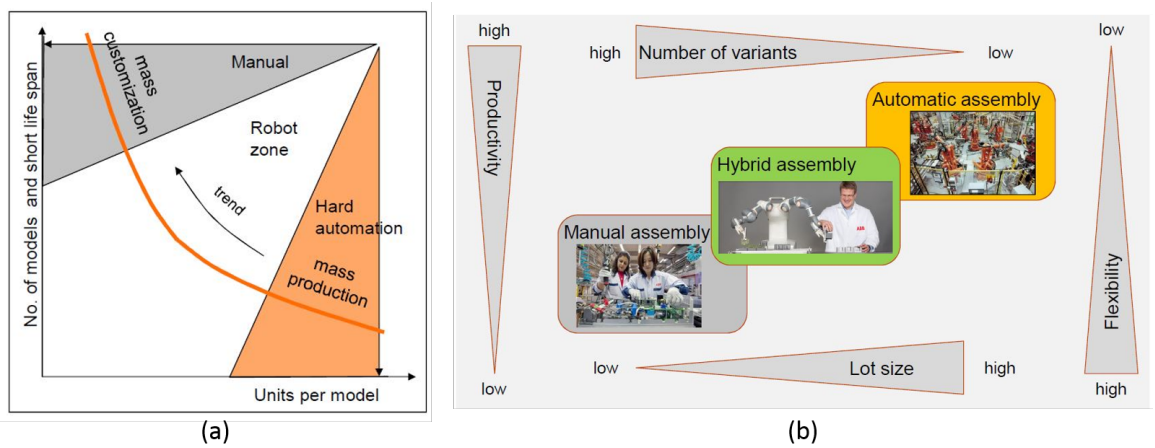


Figure 1.2: (a) A graph representing the current market trend by showing the relation between the number of units of a product model produced to that of the product life span. (b) A chart showing the balance of productivity, lot size, the flexibility of automation and variants i.e. the complexity of automation during manual, human-robot hybrid and automatic assembly [10].

collaborative robots and human-robot collaborative setups to make up for the skill gap, retain skilled task staff, attract the younger generation and thereby increase the impact on production.

From a cost perspective, Figure 1.3(a) compares the unit cost to production volume and demarcates zones of manual assembly, human-robot collaboration, robotic automation, and fixed automation. Figure 1.3(b) compares the cost of manufacturing to the degree of automation in industry. It can be observed that raising the degree of automation becomes increasingly expensive with a production process that has high variants and short lifespan. Hence the optimum degree of automation is always less than 100%. On the other hand manual manufacturing becomes increasingly competitive for remaining fraction of the production task. Hence, the process of automation of a production must involve the role of a human worker. The drawbacks of automation with the goal of reducing human error are highlighted in [36]. This work discusses the ways in which automation of industrial processes may expand rather than eliminate problems with the human operator.

> *"Yes, excessive automation at Tesla was a mistake. To be precise, my mistake. Humans are underrated,"* - Elon Musk

As quoted by Elon Musk, that excessive automation resulted in a lot of setbacks for the

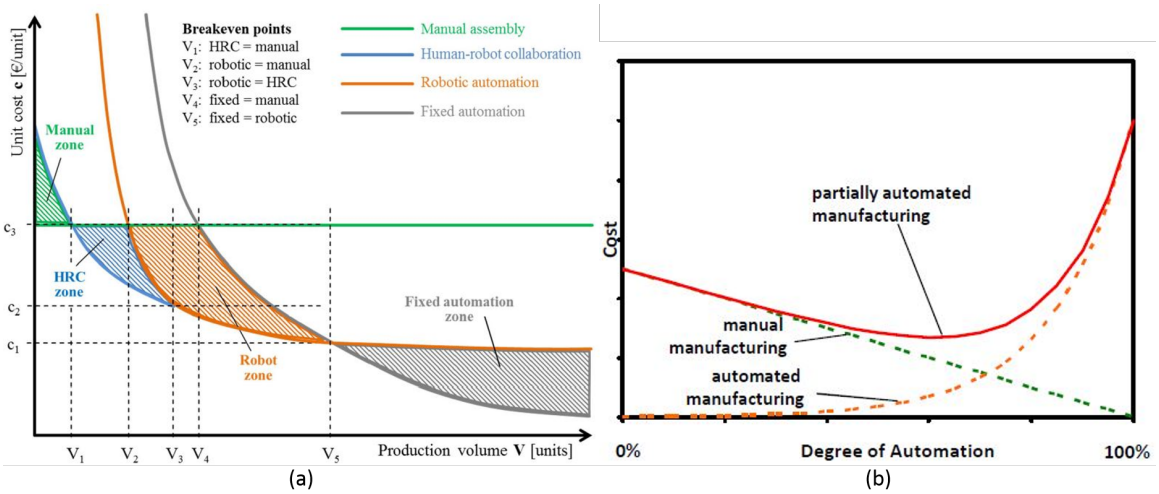


Figure 1.3: (a) A comparison of unit cost to production volume and demarcates zones of manual assembly, human-robot collaboration, robotic automation and fixed automation. (b) A comparison of the cost of manufacturing to the degree of automation in industry [10].

production of Tesla's Model3 vehicle. The technical challenge and cost of automation for some of the assembly tasks was very high in comparison to that being done manually by a human operator.

The importance of a robot in automation is to perform tasks that are otherwise dangerous, time-consuming, or outside the human desire or capability. This frees up human workers to do jobs that would require more creativity, dexterity and brainpower. This has resulted in the development of new applications in production processes that take into account the collaboration with a human operator. According to forecasts from the International Federation of Robotics, 1.7 million new robots will be installed in factories around the world and will alter operations significantly by 2020 [37].

In the next sections, we will present the research motivations for HRC and the proposed research.

1.1.2 Research Motivation

In any kind of collaborative task that involves robot sharing workspace of human-like assistive robotics in nursing homes or restaurants, collaborative assembly lines in manufacturing, or guided task manipulation; the robots have to operate in a dynamic and contextually rich environment. The robot motion, whether it is human-guided or autonomous, requires

awareness or information of the environment and the dynamic agents in its workspace. This field of research is commonly known as *Human-Robot Interaction (HRI)*. HRI is a vast field that tries to answer questions not only from an engineering perceptive but also from a societal stance. The following definition by the National Science Foundation about HRI has been quoted below [38].

> '*Human-Robot (and/or Agents) Interaction (HRI) focuses on physical, cognitive, and social interaction between people and robots (and/or agents) to project and extend human capabilities and provide for collaborative intelligence. New knowledge is needed to better understand the structural complexity of such interactions and the design of robots (or agents) that humans will find usable and useful in many contexts, such as in responding to crises, delivering health care, assisting the elderly, and increasing productivity in the workplace.*'

The HRI research presented here addresses solutions for 'increasing productivity in the workplace'. In a workspace shared by humans and robots, the main cause of injury is human-robot collision. The severity of the injury is directly related to the speed and force of the robot at the moment of collision. At the same time, the speed and force capabilities of the robot are very important for the production performance. Hence, different ways to balance these scenarios for the most optimal performance are to be researched. The objective of this proposed HRI research is a flexible and dynamic implementation of a safety measure for a human and industrial robot collaboration that optimizes the productivity of a task while ensuring human safety. Figure 1.4 shows a graph that relates the safety, performance, and productivity in the industry.

In this research, a system of systems (cyber-physical system) architecture for conducting HRC experiments is proposed and implemented. This system is used to control the robot motion, monitor human physiological response, and to track the progress of the collaborative task. This system is used for a comparative analysis of the proposed safety measures in terms of safety, performance, and productivity.

In an announcement by the NSF, there is a lot of interest in the field of *Cyber Human Systems (CHS)*. HRI, which is now considered as a subset of CHS, is an important aspect of research goals outlined by NSF for the 2016-2017 year. Given below is a goal quoted

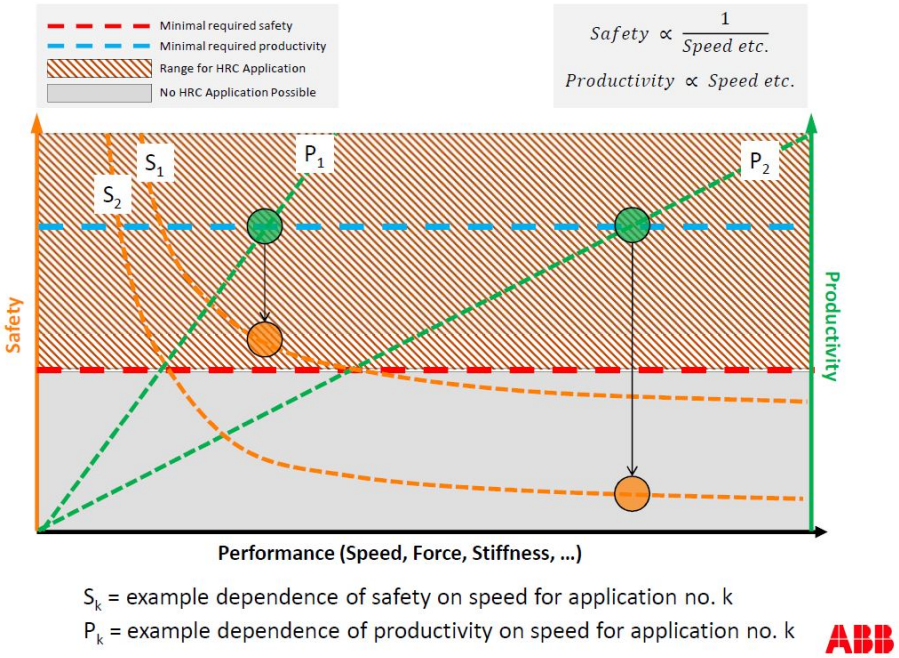


Figure 1.4: A graph that relates the safety, performance and productivity of a robot in industry [10].

from the recent NSF announcement regarding the CHS systems [39]

> '*improve the intelligence of increasingly autonomous systems that require varying levels of supervisory control by the human; this includes a more symbiotic relationship between human and machine through the development of systems that can sense and learn the human's cognitive and physical states while possessing the ability to sense, learn, and adapt in their environments.*'

This aligns with the objective and methodology of the proposed study.

Next, we present the main challenges in HRC in industry.

1.2 Main Challenges in HRC

The concept of HRC is not new, and there are many examples of HRC applications that are revolutionizing a diversity of fields in automation. A few objectives of HRC applications are bringing forth the ability for robots to perform new tasks from natural human instructions, learn new tasks from a human expert demonstration, and work with humans in the

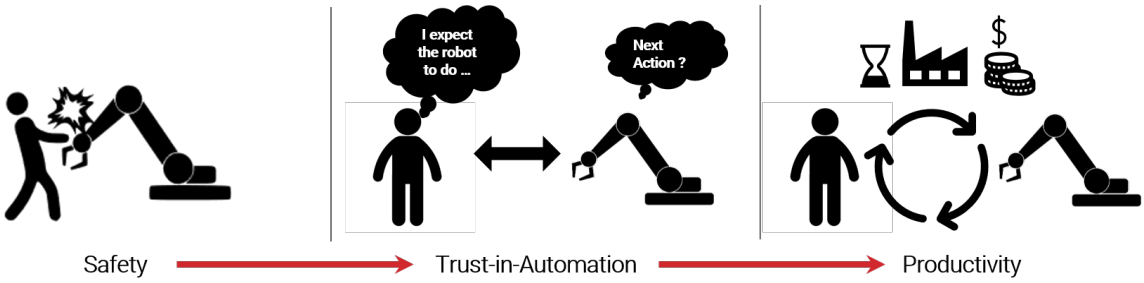


Figure 1.5: Main Challenges in Human-Robot Collaboration

same shared workspace. This applies especially to humans who are domain experts but not robotics experts [40] and are working alongside robots without hindering each others' productivity [6]. An example is a robot performing tasks collaboratively with a human teammate, where both the skills of the human and the robot can complement each other to accomplish a task that neither can achieve alone [41]. While the number of ways a robot can collaborate with a human are limitless, this type of collaboration introduces new challenges to robotics research for industrial settings and demands for a well defined industrial standards.

The three main challenges of Human-Robot Collaboration (HRC) in industry tackled in this research are : human safety, human's 'trust-in-automation' [42], and productivity [6] (representation is shown in Figure 1.5).

Human safety has always been the primary concern in robotics. One main aspect is the injuries due to human-robot collision. In order to ensure human safety and reduce risk during HRC, different risk reduction strategies have been outlined by the industrial standards for robot safety [12, 43]. One of them is the use of electronic and physical safeguards in and around the robot workspace. However, in an industrial context where productivity is paramount, overly restrictive safety measures may lead to loss of productivity.

On the other hand, relaxed safety measures can result in injuries to the human operator and loss of *trust*, which would also negatively affect the overall production. Any form of physical human-robot collision, unexpected and interrupting motion from the robot could affect the human's trust in the robot. Safety is the first step for a better human-robot collaboration and building the human operator's trust. Thus, collision avoidance strategies that change the robot motion, such as stopping and reducing operation speed while the human is in the shared workspace are critical and implemented in this proposed research.

Human *trust-in-automation* is about managing human expectations. It is about how comfortable the human is sharing the robot workspace and collaborating with the robot. The trust between a human and the robotic automation process can be achieved by managing the task-specific expectation and actions of the robot that are based on how a human operator expects the robot to behave. Another way to build trust is through effective communication between human and the robot. This could be in the form of human control or in terms of feedback from the robot. This compliance of an automation system in terms of human control and communication can help in building trust.

Safety and *trust* both positively affect the productivity of the robot [36, 42]. Efficient robot motion and anticipation of human actions and presence can also ensure faster execution of tasks. In industry cycle times are important. There is no doubt that a fully automated system would provide the highest productivity. However, as soon as the production requirements change, the flexibility of production and swiftness needs to be optimized. There is a great need for new strategies and approaches within human-robot collaboration as there are fewer standards available to implement complex protection schemes. Thus, research in understanding and analysing HRC setups that ensure human safety, build human *trust-in-automation* while optimizing the productivity are required.

As safety is one of the main objectives of any HRC setup, next, we will explore the safety measures for HRC in industry.

1.3 Safety Measures for HRC in Industry

In industry, arm robots are one of the most commonly used machinery for automation industry. Arm robots provide precise, fast, and robust solutions for automation and production. They are used to perform tasks that are otherwise dangerous, time-consuming, repetitive, or outside the desire or the capability of a human. These applications range from manufacturing, transportation, packaging, and other niche industries. The biggest advantage of robots is the increase in the overall productivity of industry due to its programmable flexibility to perform different kinds of tasks. However, the use of robots in industry increases the risk of human injury due to the lack of any inherent intelligence. This is one of the biggest reasons that robots work in safeguarded cells or have safety zones demarcated by specific sensors. The guidelines of safety measures of use of robots in industry

have been standardized by the International Standards Organization (ISO) [12, 43]. Arm robots (or robot manipulators) are being commercially made and designed with these standards kept in mind. They have a robust control architecture that ensures the performance of commanded actions by the operator or the program written by the operator. Installing these robots also require standards to be met to ensure a safe working environment for human operators around it. The reason being, there is no rational stand alone intelligence embedded in the robots.

With the onset of Human-Robot Collaboration setups in industry, the current safety standards can become overly restrictive. Currently, the two common approaches used for HRC in industry are the use of collaborative robots and electronic safeguarding measures.

First is the development of human-complaint robots. Compliance is achieved by changes in both hardware and software implementation of the robot. Hence, a new category of robots called *collaborative robots* or *cobots* have been introduced in the market. These robots are purposely designed to work in direct cooperation with humans in a defined workspace. This is achieved by incorporating protective force stops when it detects a collision and back drivable motors that allowed humans to interrupt the robot movement without worrying about being injured. This is achieved by decreasing the overall momentum of the arm by sensing the resistance force (force sensing) in the actuators of the robots itself. The robot used in this research is Universal Robot UR-series cobot, which is one of the most used cobot for many industrial applications [3]. The UR-series cobots UR3, UR5, and UR10 are shown in Figure 1.6.

The second approach is to make the robot aware of its surroundings and change its motion to prevent human-robot collisions. For these safety approaches, physical and/or electronic safeguarding measures have been implemented. Electronic safety devices are the sensors currently being used in industry to demarcate safety zones around a robot work cell to reduce the risk of human injury due to human-robot collision. A brief description of these sensors is given below (also refer Figure 1.7) :

(1) **Safeguarded perimeter door switches** are used when a robot work cell has physical safeguards such as rails around them. These switches are used to enable preventive stops if a door that allows access to the robot work cell is opened.

[3]The UR robots have a 60% global share of the cobot market, selling more cobots than all competitors combined. URL

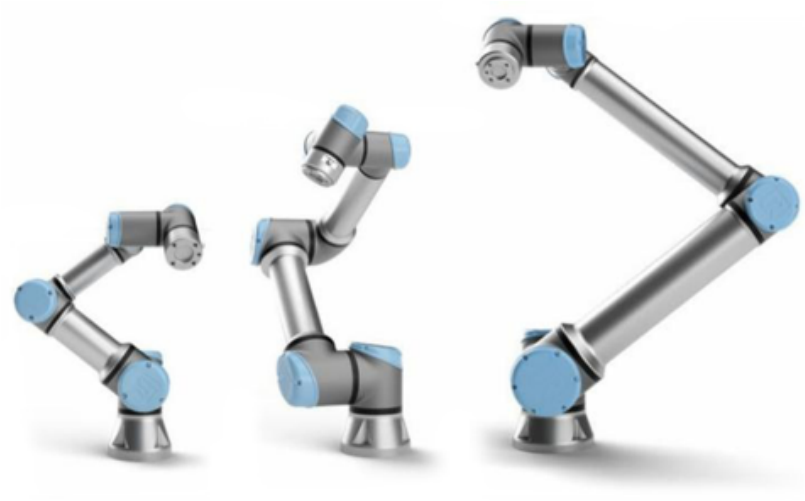


Figure 1.6: Universal Robots UR-series cobots [11].

(2) **Light curtains** are electronic laser curtains that act as virtual outlines of a perimeter around the robot work cell. They detect the breach of a plane when the perimeter of the robot work cell is breached.

(3) **Pressure pads/floors** are used to detect the presence of human operator(s) by sensing pressure applied to the floor. These are commonly used on the floor space around the robot.

(4) **2D scanning LiDAR** is used to measure distance in a plane between a human and a robot. It is used as a proximity sensor in the industry. The distance thresholds are task-specific.

(5) **Safety Vision Scanner** is a 3D vision sensor used to demarcate safety zones in three-dimensional space as shown in Figure 1.7.

In this research, we have developed a Time-of-Flight (ToF) based proximity sensor as an electronic safeguarding measure using the Speed and Separation Monitoring (SSM) methodology, which is used conventionally with 2D LiDARS in HRC setups as outlined in the industrial standards. This safety measure was used with a UR10 cobot to ensure human-safety when validating and testing the SSM safety measure. The intended application of the ToF based SSM setup is independent of the type of robot used. The SSM safety measure is discussed further in the following section.

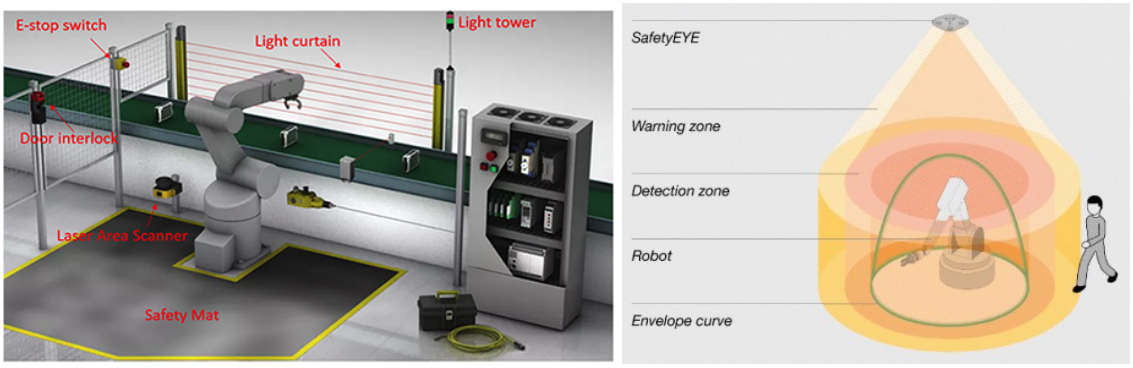


Figure 1.7: Physical and electronic robot safety measures currently used in industry. (Image Source: DigiKey, PilZ)

1.4 Speed and Separation Monitoring as a Safety Measure

Different strategies have been introduced to ensure human safety and are outlined in the industrial standards [12,43]. One of the ways of maintaining the safety of a human operator during HRC is Speed and Separation Monitoring (SSM). In SSM, based on the minimum separation distance and relative velocities between the robot and the human, the robot stops or slows its motion to avoid a collision. The conventional setups of SSM in industry use 2D scanning LiDARs to measure the distance between robot and human. Here, safety zones around the robot are demarcated using fixed distance thresholds.This is termed as 'static awareness' of the robot. The Figure 1.8 shows a basic SSM setup in a simulation representing static 2D safety zones around the robot.

Such setups are inefficient in cases where the human overlaps with robot operating workspace more often, resulting in frequent robot halts, thereby negatively impacting the overall productivity. Therefore, a more flexible and dynamic implementation of safety measures that optimizes the productivity of a task while ensuring human safety is needed. Unlike the static/fixed demarcated 2D safety zones, a sensor setup that changes the safety zones based on the robot pose and motion is presented in this dissertation. Such safety measures are called *Dynamic Speed and Separation Monitoring* [18, 44].

For any human-robot interaction to occur, the robot must have the information associated with its environment via exteroceptive sensors [45] such as LiDAR(s) [46], cameras [44,47], and radars [48]. The placement of these sensors in the environment determines the workspace area coverage of the robot. Using sensors mounted on the robot can provide

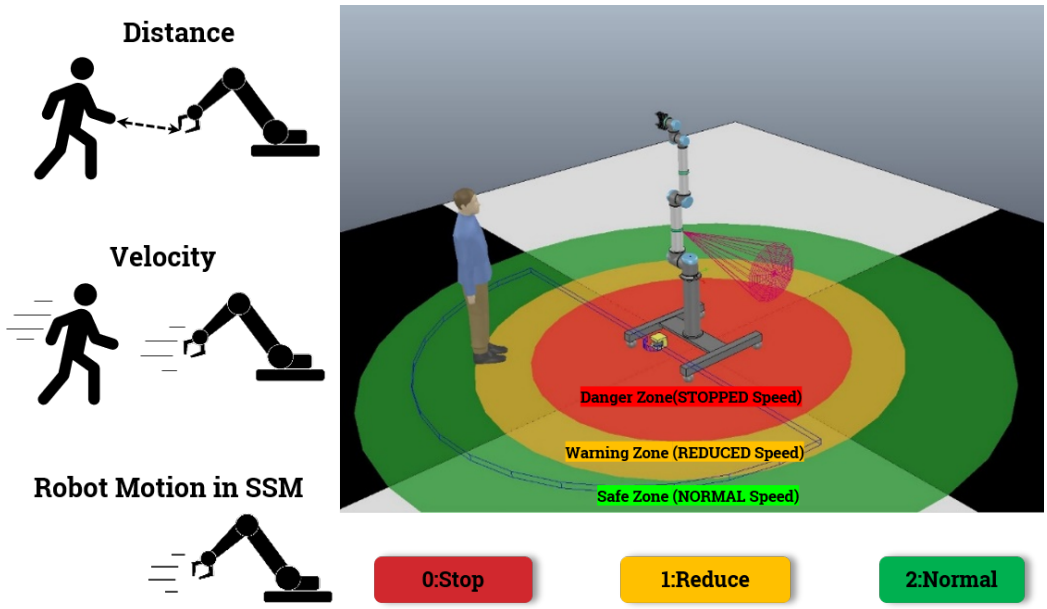


Figure 1.8: A basic SSM setup in a simulation representing static 2D safety zones around the robot. There are three safety zones, *Danger*, *Warning* and *Safe*, where the robot motion *stops*, *reduces* speed (slows down), or *normally* moves, respectively, to ensure the safety of the human [10, 12].

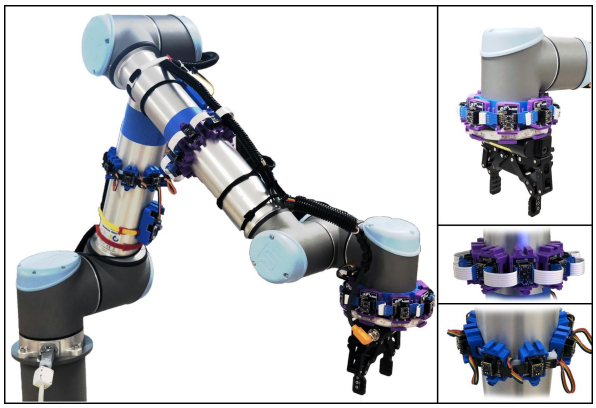
information from its perspective while removing the constraints of planning the placement of the sensors in the environment [1, 19, 49]. They can also give direct observations without the need to apply transformations to elicit relevant distance information associated with the human. This research compares the dynamic SSM safety configurations based on ranging sensors placed on the robot and vision sensors placed around the robot for monitoring the collaborative workspace.

Next, we present the research statement and roadmap of this dissertation.

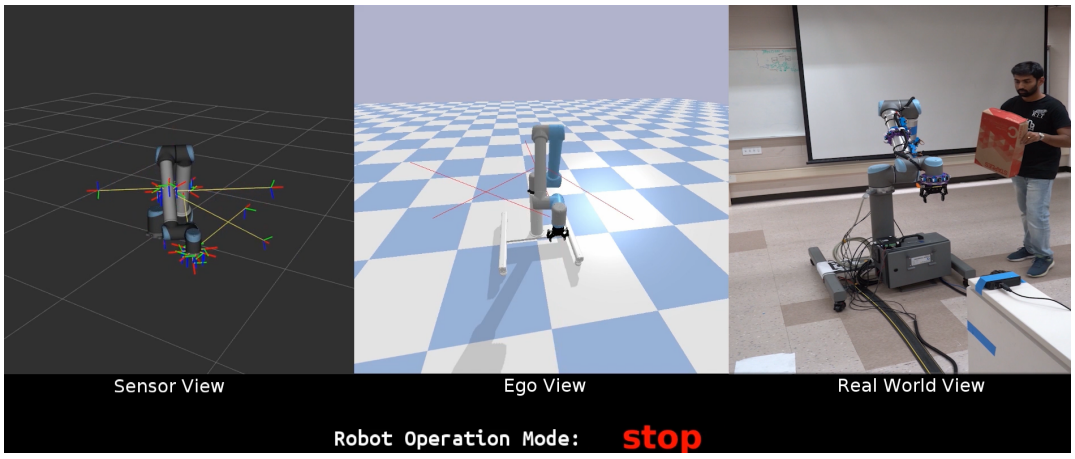
1.5 Dissertation Research Statement and Roadmap

The focus of this dissertation is the implementation of a Speed and Separation Monitoring safety configuration using Time-of-Flight (ToF) laser-ranging sensor arrays placed at the centers of the links of a robot arm. This work investigates the viability of on-robot exteroceptive sensors for implementing SSM as a safety measure [1, 2, 4, 6].

The figures below show the hardware and software implementation of dynamic SSM using a ToF sensor array prototype.



The UR10 robot with ToF sensor arrays placed at the centers of robot links. Each array has eight single unit lidar(s).



A Dynamic SSM Safety Measure implemented for a UR10 robot with ToF sensor arrays placed at the centers of robot links Video Link

.

The organization of the research in this dissertation is as follow:

Chapter 2 provides background information and a literature review of related research. It presents a brief summary of the existing industrial standards for the use of robotic automation and HRC. This chapter positions the work reported in this dissertation in the Human-Robot Collaboration Taxonomy. It presents a detailed categorization of Human-Robot Collaboration based on a conceptual framework. This survey was done as part of an independent study to explore the current HRC research in industry.

Chapter 3 explores the components and sensors needed for the implementation of a dynamic safety measure, such as dynamic SSM. Here, a framework for the adaptive adjustments of robot behavior using real-time perception as a system of systems (a cyber-physical system) is presented. The pros and cons of using exteroceptive sensors such as the RGB-D cameras (Microsoft Kinects) and a Motion Capture system for implementing SSM safety setup is discussed. This chapter discusses the ideation of the concept of using on-robot ranging sensors for safety. The approaches to use simulation tools for a *digital-twin* interface for control and monitoring of the robot and its motion for safety are also detailed.

In Chapter 4, the main contribution of this dissertation is presented. The first part of this chapter shows the validation of ToF sensor arrays as on-robot ranging sensors for implementing a dynamic safety measure by using a *digital-twin* simulation setup. Here, the robot behavior is simulated by *hallucinating* a human working in the robot workspace. Algorithms for safety and robot motion are formulated. The results are evaluated in terms of safety, performance, and productivity and compared with a 2D LiDAR scanning safety configuration. The second part of Chapter 4 shows the implementation of different dynamic SSM safety configurations based on approaches of measuring human-robot separation distance and relative speeds using the sensor modalities of ToF sensor arrays, a motion-capture system, and a 2D LiDAR. A comparative analysis of the proposed dynamic SSM safety configurations in terms of its safety, performance, and productivity is presented.

Chapter 6 presents the design and implementation of the prototype for an 8-node ToF sensor array used for the application of dynamic SSM in this research. It highlights the performance and drawbacks of this prototype, *ToF Prototype V1*. For addressing the disadvantages of the *ToF Prototype V1*, the chapter presents details of a second prototype, *ToF Prototype V2*, which is also a minimum viable product with a modular design and up-to 32 sensor nodes per array. This chapter discusses the results of a sensor characterization

analysis in terms of accuracy, overlap coverage, and ranging under motion. It also provides details of the hardware and software implementation of the prototypes and the interface of ToF sensor arrays on the robot for dynamic SSM. By using the *digital-twin*, the placement of the ToF sensor arrays on the robot links and their sensing volume coverage is analyzed. A methodology for volumetry using octrees to quantify the detection/sensing volume of the sensor arrays is described in this chapter. The change in sensing volume coverage by increasing the number of sensors per array and the number of arrays placed on the robot is studied, and its result tabulated.

The human subject experiment setup for a standard material handling pick and place robot task and a human assembly task done in a shared collaborative space are described in Chapter 5. Experiments with human subjects under the proposed SSM safety configurations were performed, and the subjective feedback in-terms of the human's *perceived* safety, preference, and comfort is quantified in this chapter. These results of the experiments are presented in this chapter.

Conclusions are drawn, and future work is discussed in Chapter 7. The contributions are summarized, and the impact of this work on other research is presented.

Chapter 2

Background and Related Work

This chapter provides background information and a literature review of related research on safety measures for HRC in industry. A brief summary of the existing industrial standards of the use of robots and collaborative robots in industry is presented. The work reported in this dissertation is positioned in the Human-Robot Collaboration taxonomy. Following these sections, a detailed categorization of Human-Robot Collaboration based on a conceptual framework is presented. This was done as part of an independent survey with the goal of understanding the current HRC research in industry. Here a literature review of relevant and similar works relating to dynamic safety measures in HRC has been presented.

2.1 Safety Standards in Industry for Robot Applications

This section presents a brief summary of existing safety standards for applications of industrial robots and its related standards and directives. It also explains the safety functions of industrial robot controllers. Following which the new technical specifications and guidelines for human-robot collaboration with collaborative robots are summarized.

The industrial standards for safety of human operators with robotic applications have been evolved from safety standards with machinery. As robots have been increasingly used in industry and as they become more autonomous, faster, precise, and stronger, the safety of operators along with the applications has become an integral part of the industry. This required standardization and guidelines for the use of robots in industry to ensure the safety of human operators. This was achieved by the regulations defined by various safety standards for the use and manufacturing of industrial robots. In the United States of America, Occupational Safety and Health Association (OSHA) is responsible for outlining the guidelines for the use of industrial robots and robot systems. ANSI/RIA R15.06-2012 standard stipulates the guidelines for the use of robots in industry. Similar standards are prevalent

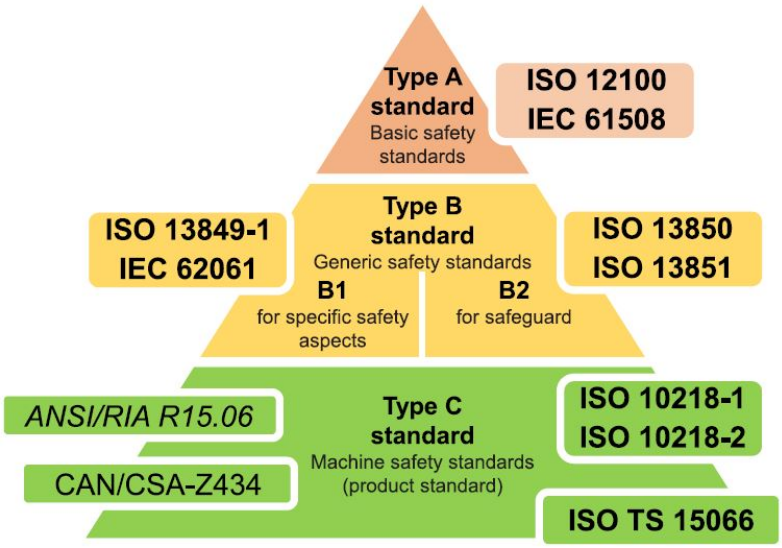


Figure 2.1: A chart of Safety Standards and Directives for Applications of Industrial Robots [13]

in other countries as well, such as the Canada Standard Association and EU Standards Organizations (CEN). All these standards are derivative of the main safety standard by the International Organization for Standardization (ISO) for applications of industrial robot, ISO 10218 [12]. ISO 10218 consists of two parts: Part 1 - Industrial Robots and Part 2 - Robot Systems and Integration. Figure 2.1 shows a relationship between various standards and their level of application [16]. The C-level standard dictates the higher level behavior of machinery in interaction with environment and humans. Here only the main directives outlined by the ISO 10218-1/2 that pertain to this research will be mentioned to give reader a general idea of its purpose [13].

The scope of requirements and guidelines outlined by ISO 10218-1 [12] relate to the functionality and characteristics of the robot manipulator - such as the inherent safe design of the robot and their controllers and information of use in industry. It describes basic hazards associated with robots and provides requirements to remove or reduce risks associated with it. Whereas, the scope of ISO 10218-2 covers the hazards that are presented by industrial robot systems when they are integrated and installed in industrial cells or lines. These hazards are unique to the robot application and relate to the nature of automation and complexity of installation. This part of the ISO standard describes possible types of risks associated with the type of robot used and its purpose and the way it is installed,

programmed, operated, and maintained.

Next, we present the types of safety stops and safety functions of industrial robot controllers as defined in ISO 10218-1 [10, 12]. The essential safety-related function for industrial robot controllers are classified based on the safety stops, operation modes, and direct control using teach pendants. There are two main categories of safety stops: *emergency stop (E-Stop)* and *protective stop*. A robot protective stop can be further categorized as follows:

- **Cat-0** requires immediate removal of power to the actuators. (sometimes considered as an uncontrolled stop because motion can take some time to cease because the motor may be free to coast to a stop.)
- **Cat-1** requires that power is retained to apply braking until the stop is achieved and then remove power to the actuator.
- **Cat-2** allows that power need not be removed from the actuator.

There are three different robot operation modes: *automatic*, *manual*, and *manual at high-speeds*. Each of the operation modes has features and constraints under which the robot operates. During manual control, a teach pendant is used as a direct interface to the robot. It must have an *enable*, *start/restart*, and *hold-to-run* switches according to the safety standards. The robot controllers also have safety supervision functions. According to the ISO standards [12], every industrial robot must have basic supervision of a robot that ensures that the robot motion executed corresponds to the motion commanded. It should also have supervision of kinematic quantities, such as

- position control of joints and end effector or the tool center point (TCP),
- speed control of the joint speeds (degree/second) and TCP speed,
- acceleration control and braking.

The other safety functions a robot controller can have is the supervision of the robot's dynamic quantities, which is controlling the torques and forces of the robot joint motor (this is a must for collaborative robots). It can also have application-related user-defined functions for control, such as gesture-based commands, voice commands, and camera-based supervision.

These characteristics of the robot drive the evolution of safe human-robot collaboration and interaction. Figure 2.2 aptly shows how the need for human-robot collaboration has resulted in the evolution of safety concepts and industrial standards for industrial robots.

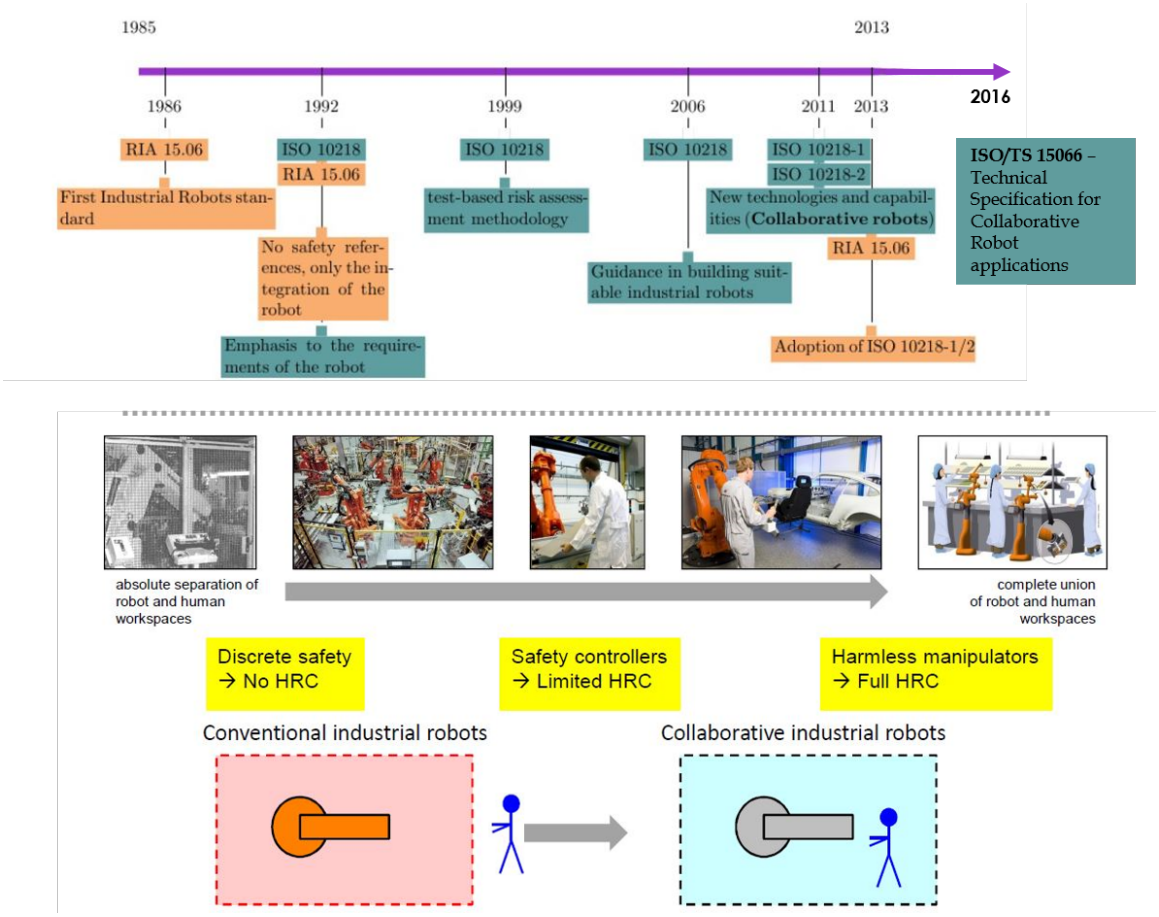


Figure 2.2: Evolution of Safety Concepts for Human-Robot Collaboration [14].

Different from the conventional robots, the collaborative robots were inherently built with human safety being the primary goal. The ISO/TS 15066 technical specification provides more information regarding the use of collaborative robots. A brief overview of this is given in the following section.

2.1.1 Technical Specification ISO/TS 15066 - Safety of Collaborative Robots

The ISO/TS 15066 technical specifications introduce a lot of data, calculations, and methodologies that have been developed to make sure the collaborative robot application is safe for use alongside humans. This specification is a precision on the existing ISO 10218 safety

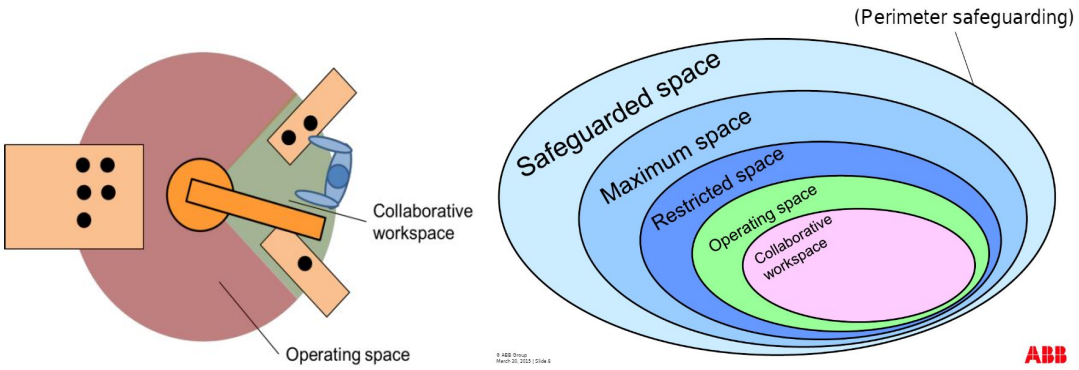


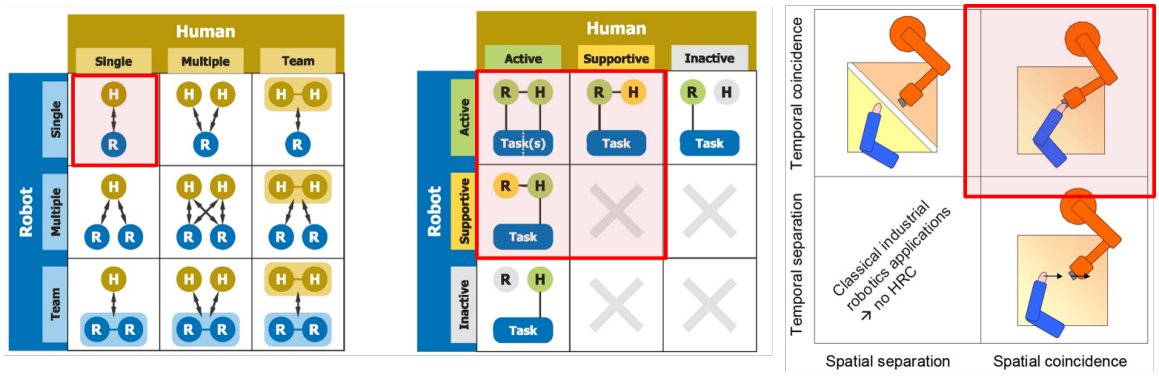
Figure 2.3: A robot cell workspace in industry [10].

standards that directly relates to collaborative robots. This specification clarifies four different types of collaboration: Safety Monitored Stop (SM), Hand Guiding (HG), Speed and Separation Monitoring (SSM), and Power and Force Limiting (PFL). These types of collaboration are present in the ISO 10218 but the new technical specification document clarifies points particularly concerning the maximum speeds, maximum pressure and force values that are allowable to achieve a safe human-robot collaboration [43]. These specifications and some definitions are presented below to give the reader an idea of collaborative operation of robot.

Collaborative Workspace and Collaborative Operation

According to ISO 10218-1:2011(clause 3.5), collaborative workspace (CWS) is *"a workspace within the safeguarded space where the robot and a human can perform tasks simultaneously during production operation"* [10, 12]. For a robot arm, generally a collaborative workspace is a subset of the robot operating space. These have been shown in Figure 2.3. According to ISO 10218-1:2011 (clause 3.4), a collaborative operation can be defined as *"a state in which purposely designed robots work in direct cooperation with a human within a defined workspace"* [10]. A robot cell in an industry can be grouped into the following workspaces:

- **Safeguarded Workspace**: A safeguard workspace is the boundary of the robot cell. This is generally where perimeter guarding using physical safety guards is done.
- **Maximum Workspace/Envelope**: Maximum workspace is defined as the maximum reachable workspace by the robot along with an end effector attached to it.



(a) (b) (c)

Figure 2.4: (a) HRC classification based on multiplicity during interaction of, (b) autonomy and initiative, and (c) spatial and temporal overlap [15, 16]. The highlighted region shows the classification of this research.

- **Restricted Workspace/Envelope**: A subset of the maximum workspace of the robot that is restricted or the robot cannot access it due to immovable objects such as the wall or table, or due to pre-defined motion restrictions.
- **Operating Workspace/Envelope**: This is the workspace where the robot moves to perform a given task.

Next, we position the research presented in this dissertation in the HRC taxonomy discussed in the literature.

2.2 Human-Robot Collaboration Taxonomy

According to [15], HRC can be classified based on the multiplicity during interaction. As shown in Figure 2.4(a), an HRC scenario can be an interaction with a single human, single robot, multiple humans, multiple robots or teams of humans and robots. In this research, the work is limited to single human single robot interaction. Another form of classification is based on the autonomy and initiative of an agent-human or robot during an HRC task. It defines the role of a human or robot during the task as active, supportive, or inactive, as shown in Figure 2.4(b). An *active role* is when the agent has the autonomy and initiative to do the task. A *supportive role* is assisting the active agent in performing the task. In this research, the HRC tasks used for analysing the SSM setup has a human and a robot in both active and supportive roles.

According to [10, 16], the HRC can be classified based on the spatial and temporal coincidence of human and robot workspace for single human-robot team collaboration, as shown in Figure 2.4(c).

Next, we present the taxonomy based on the different levels or degrees of collaboration.

2.2.1 Levels of Human-Robot Collaboration

The levels or degree of HRC operations in industry are shown in Figure 2.5. A level of HRC is based on the intelligence of the system and complexity of the task during human-robot interaction in a shared workspace. The most commonly used collaborative operations are of type *coexistence* and *sequential collaboration*. The *coexistence* level of HRC described a setup where the human and robot work in a shared workspace without a physical safety guard. For an HRC setup of *sequential collaboration*, the robot and the human subject share a collaborative workspace, but the movements are sequential. With the increasing level of collaboration, the need for the use of external sensors for providing safety during HRC also increases. The SSM based safety configuration is most commonly used for *sequential collaboration* level of collaboration.

The HRC setups can also be classified based on the levels of protection and protection schemes used for safety and as risk reduction measures to collision in an HRC setup in industry. These are described in the following section.

2.2.2 Protection Levels for Risk Reduction

The solutions to provide safety for human operators are classified into different protection levels. Figure 2.6 shows the different levels of protection for human operators working in the shared human-robot workspace. *Levels 1, 2,* and 3 are the most commonly implemented protection levels in industry. *Level 5* is generally application-specific, such as a robot performing metal welding tasks. *Level 4* and *6* are currently the key areas of this research. The need for Level 4 protection has resulted in the development of collaborative robots, such as Rethink Robotics Baxter, Universal Robot UR3/5/10), and Kuka LWR iiWA. It has also inspired research in physical human-robot interaction that deals with implementing collaborative behavior in industrial robots. It should be observed that the degree of collaboration

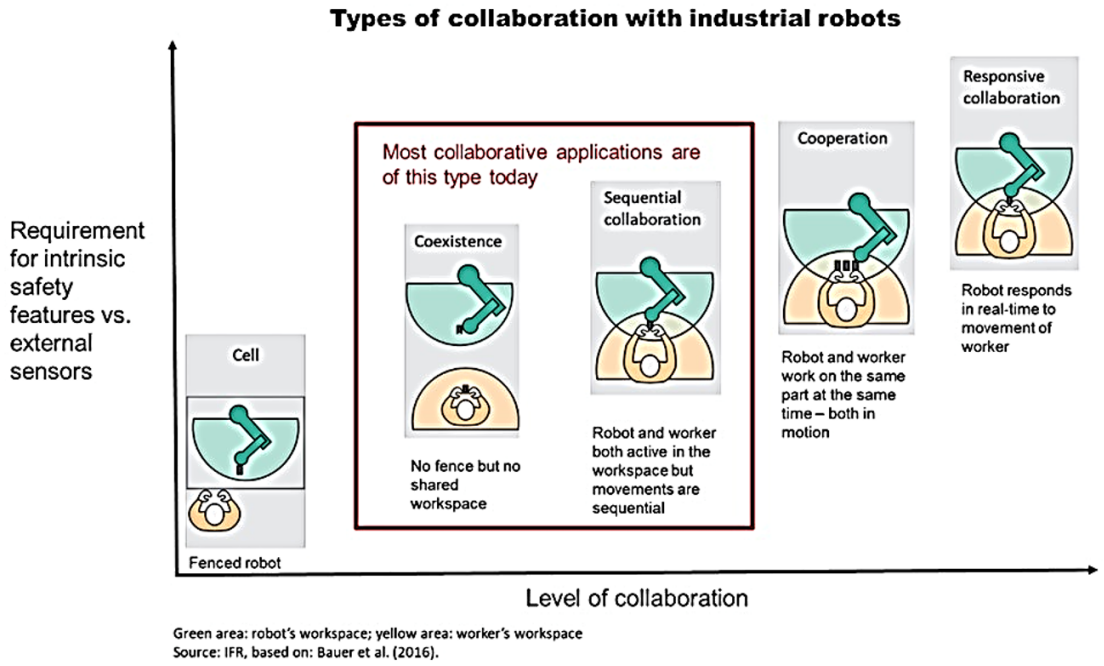


Figure 2.5: A diagram showing levels of human-robot collaborative operations in industry [17].

increases with the level of protection. The proposed research attempts to implement a system that would provide a *Level 6* protection by monitoring the real-world environment and adjusting the robot motion to the changing environment and human motions.

The types of collaborative operations outlined in ISO 10218 [12] that can be performed with the above mentioned protective schemes are described in the next section.

2.2.3 Types of Collaborative Operation

There are mainly four types of collaboration outlined in the initial ISO 10218-Part 1(2011) standards. A table showing the suggested risk reduction measures for these types of collaboration is shown in Figure 2.7. They are described as follows [12]:

(a) **Safety-rated Monitoring Stop** - In this collaborative operation, when the worker is in the CWS, the robot is not allowed to move. When sharing the workplace no contact between the robot and the human operator or worker is permitted.

(b) **Hand Guiding** - The worker has absolute control of the robot and the robot is only moved when it is activated by the operator. The robot speed is limited.

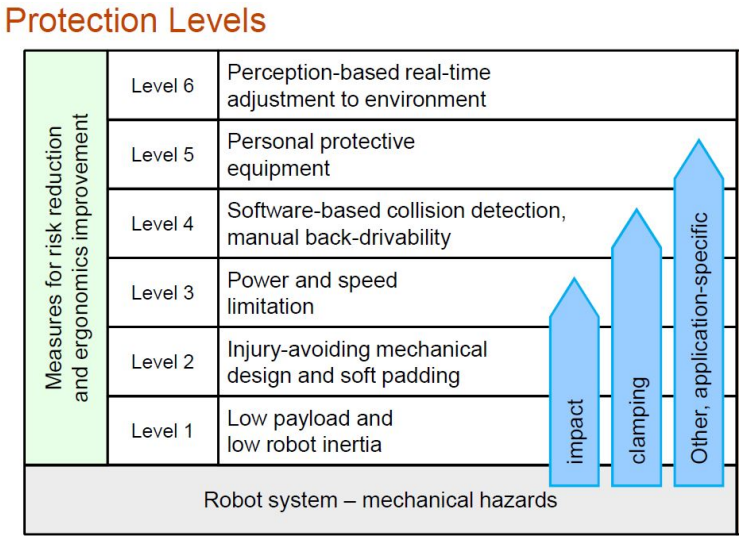


Figure 2.6: The different levels of protection for human operators working in the shared human-robot workspace [10].

(c) **Speed and Separation Monitoring** - The worker position is supervised and the speed and position of the robot is adapted in order to prevent contact. A minimum separation distance is maintained between the human and the robot.

(d) **Power and Force Limiting** - The robot is able to detect a collision quickly, take a remedial action to minimize the impact and harm to the human and the robot, and recover post-collision. An intentional human contact can be detected, and the force applied can be used as a form of direct control to move the robot links. This mode of control for collaborative robots is called the 'free-drive' mode [11]. This field of research is called physical human-robot interaction (pHRI) [18, 50, 51].

The types of HRC contacts that can happen at the moment of human-robot collision take into consideration the contact and impact between a human and a robot. These contacts are classified as *Transient* contact, also known as *Free Impact*, and *Quasi-Static* contact, also known as *constrained impact*. The differences are shown in Figure 2.8. In SSM, any form of contact is avoided, but it is implemented mainly for avoiding transient contact.

The HRC setups can be classified based on the multiplicity or the autonomy of agents, the level or degree of collaboration, protection levels, and types of risk reduction measures. To summarize, the dynamic SSM setup presented in this research is limited to a single human single robot interaction where the human and robot are in both active and supporting

Types of Collaborative Operation According to ISO 10218-1

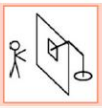
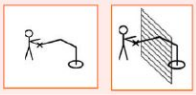
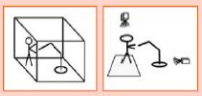
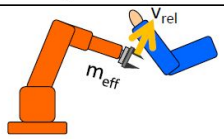
ISO 10218-1, clause	Type of collaborative operation	Main means of risk reduction	Pictogram (ISO 10218-2)
5.10.2	Safety-rated monitored stop (Example: manual loading-station)	No robot motion when operator is in collaborative work space	
5.10.3	Hand guiding (Example: operation as assist device)	Robot motion only through direct input of operator	
5.10.4	Speed and separation monitoring (Example: replenishing parts containers)	Robot motion only when separation distance above minimum separation distance	
5.10.5	Power and force limiting by inherent design or control (Example: *ABB Dual-Arm Concept Robot* collaborative assembly robot)	In contact events, robot can only impart limited static and dynamics forces	

Figure 2.7: A table showing different types of collaborative operation in industry [10]

	Transient Contact	Quasi-Static Contact
Description	• Contact event is "short" (< 50 ms) • Human body part can usually recoil	• Contact duration is "extended" • Human body part cannot recoil, is trapped
Limit Criteria	• Peak forces, pressures, stresses • Energy transfer, power density	• Peak forces, pressures, stresses
Accessible in Design or Control	• Effective mass (robot pose, payload) • Speed (relative) • Contact area, duration	• Force (joint torques, pose) • Contact area, duration



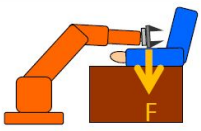


Figure 2.8: A table showing different types of Contacts between Human-Robot [10]

roles. The types of experiments used in this research implement a *sequential collaboration* HRC scenario for speed and separation form of risk reduction safety measure.

In the next section, we present a conceptual categorization and survey of the HRC research in industry.

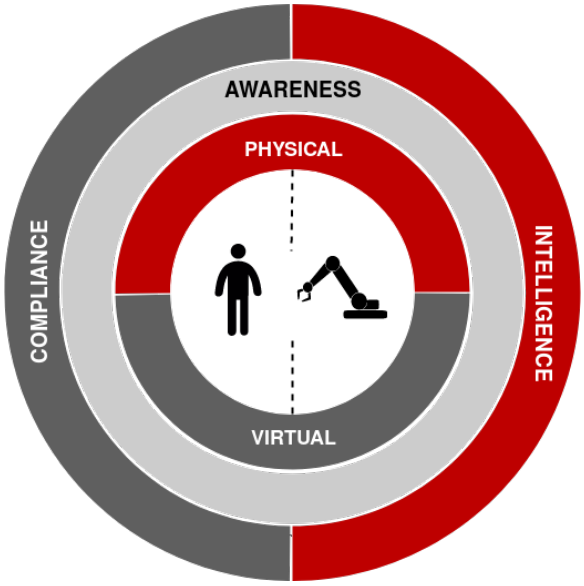


Figure 2.9: A system diagram for the proposed conceptual framework for a Human-Robot Collaboration as System of Systems (HRC-SoS). It highlights the three aspects of an HRC system: AWARENESS, INTELLIGENCE and COMPLIANCE [5, 6].

2.3 Conceptual Categorization of HRC in Industry

The research for Human-Robot Collaboration (HRC) setups in the industry is done to ensure human safety, build human trust in automation while optimizing productivity. With that objective in mind, in this section, a Human-Robot collaboration framework as a System of Systems (HRC-SoS) is presented. Here, the HRC setup can be conceptually categorized as consisting of three main aspects: AWARENESS, INTELLIGENCE and COMPLIANCE [6]. These aspects are discussed in detail in further sections, and related HRC research is categorized based on these aspects and their corresponding subsystems [5]. A conceptual system diagram for the proposed HRC-SoS is shown in Figure 2.9 and an outline of the HRC survey is shown in Figure 2.10.

AWARENESS aspect addresses the level of perception using sensors on the robot, human, or in the workspace. It is the information that represents the state of the human-robot collaboration. AWARENESS aspect can be categorized based on the source of information: *physical or real world* information, *virtual or digital* information generated in the simulation or as a result of the combination of *physical* and *virtual* information.

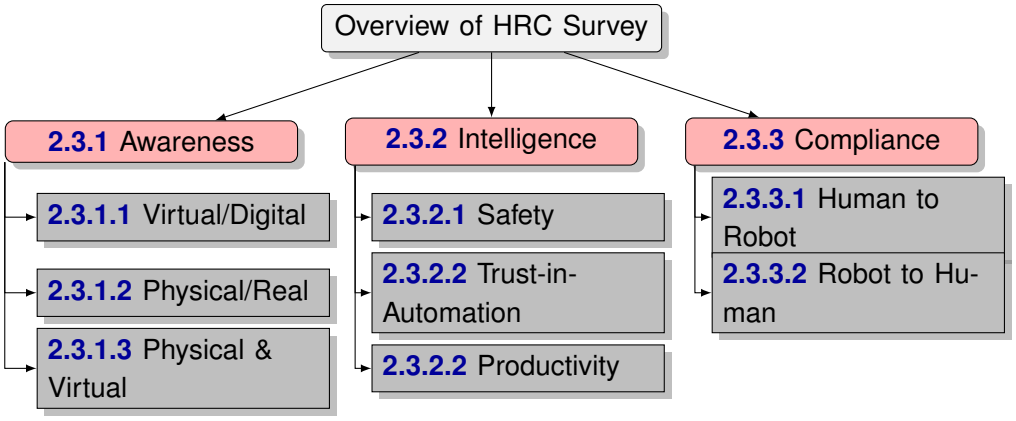


Figure 2.10: A tree diagram depicting the outline of the HRC survey, as discussed in Section 2.3.

INTELLIGENCE of HRC is about robot actions and behavior. The algorithms and logic to achieve the objective of the HRC is its INTELLIGENCE. The INTELLIGENCE aspect can be loosely categorized based on the desired change i.e. the objective and intended application of the HRC setup. The main objectives (challenges addressed) are addressing either the *safety*, *trust-in-automation*, and *productivity*, or all of them during HRC.

COMPLIANCE is about the robot managing human expectation, human control over the robot and efficient communication and feedback between a human and a robot. COMPLIANCE nature can be categorized as *human-to-robot* and *robot-to-human*.

In this section, the relevant and similar research that has influenced and inspired the research presented in this dissertation is categorized based on the HRC aspects mentioned above. The AWARENESS aspect of the proposed HRC system provides a categorization of various methodologies used for physical and virtual information to represent human-robot collaboration. The INTELLIGENCE aspect covers the relevant research pertaining to safety in HRC. It further expands on the research relating to collision control methodologies based on Speed and Separation monitoring applications in HRC. The COMPLIANCE aspect of the HRC system is explained briefly with references to applications in research and industry.

2.3.1 Awareness Aspect of HRC

In an HRC setup, the AWARENESS represents the information regarding the human, robot, objects of interest, and the workspace. This has been further categorized based on the

source of information. The *physical* information is generated as a result of sensors in the workspace (e.g. cameras, 2D LiDAR), on the human (e.g. bio-metric sensors), robot (e.g. encoders, force-torque sensors and distance ranging sensors) and other peripherals (e.g. buttons and pressure pads). The data generated from the sensors in the physical world enables the perception during HRC. The *virtual* information is generated through computer-based simulation. The physical and virtual information can be combined and used in HRC to represent the state of the human, the robot, and the workspace. The examples are the use of augmented and mixed reality (AR/MR) [52] and virtual reality [53] for human-centric interaction in industry. Another example that is used the most is *digital-twin* also known as a *simulation-in-the-loop* [54, 55].

2.3.1.1 Virtual/Digital Awareness

Virtual/Digital awareness of an HRC system is information generated in a computer-based simulation. Simulation is used extensively in industry for robot motion planning, designing production setups, factory automation, cycle times and productivity logistics. This is used to validate and test scenarios of robot behavior. Some of the commonly used simulation softwares used in industry are Robot-DK, Visual Components, Microsoft Robot Studio, Energid-Actin, Process Simulate, and V-REP (a.k.a. CopelliaSim). The simulation is used as an ideal representation of the physical world. The information generated in the virtual world can be used to test and develop HRC concepts and setups.

2.3.1.2 Physical/Real World Awareness

Sensors result in perception, and perception defines the Awareness of the robot, the human in the HRC setup. The limitations of the sensors dictate the level of Awareness in an HRc setup. The perception can be further categorized as *proprioceptive* and *exteroceptive* sensing [45]. An outline for the categorization of physical world information has been shown in Figure 2.11.

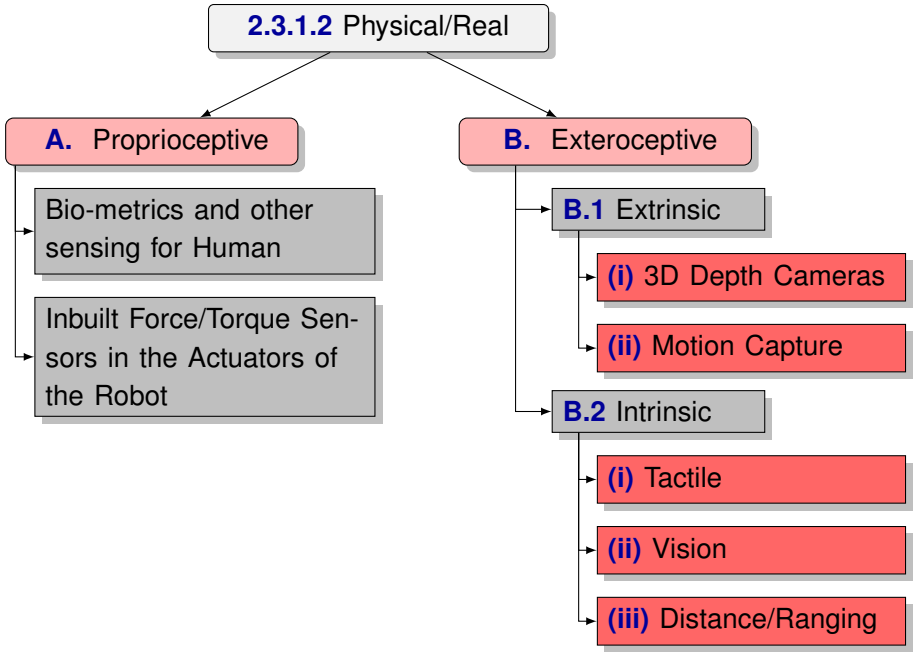


Figure 2.11: A tree diagram depicting the categorization of Physical/Real World Awareness in HRC, as discussed in Section 2.3.1.2.

A. Proprioceptive Sensing

Proprioceptive Sensing can be defined as 'relating to stimuli that are produced and perceived within a body, especially those connected with the position and movement' [1]. The examples of proprioceptive sensing for humans are bodysuits with Inertial Measurement Unit (IMU) and bio-metric sensors that generate data as a result of human motion or action. In [55], an IMU bodysuit was used to track the body movement and pose. The human pose, in conjunction with 2D LiDAR was used for an SSM safety setup. Bio-metric sensors for electromyogram (EMG) or signals from muscle movement have been used as a form of control and information in HRI. The work in [56] shows how EMG was used to measure fatigue during human-robot collaboration. The result was that the detected loss of effort due to human fatigue was compensated by the robot. Other bio-metric sensors for measuring heart rate, respiration rate, galvanic skin response (GSR), pupil dilation and eye movement for gaze tracking can be considered proprioceptive senses and have been used during Human-Robot Collaboration [18, 57, 58].

[1] a dictionary definition of the word 'proprioceptive' Link

For robots, the proprioceptive sensing is the internal information such as the encoders for measuring joint angles, current and voltage measurement for each joint, force-torque sensors and the internal temperature of the motors. The most commonly used information is the kinematic and dynamics of the robot that is determined using the joint angles, joint velocities, and joint torques. This information defines the motion, pose, and behavior of the robot. In this research, this information is used to generate a *digital-twin* of a robot in 3D simulation and mimics the robot movement according to the information provided. The force-toque sensors built-in with the motors have resulted in compliant actuators for robot-joints that have led to the development of collaborative robots. For safe physical HRI, this data has been used to determine the differential change in in torques (also called *rotatum*, indicative of the jerky motion of the robot) to trigger safety stops in robot [50, 59].

B. Exteroceptive Sensing

Exteroceptive Sensing can be defined as 'relating to stimuli external to the body' [2] which is perceiving the environment. For humans, exteroceptive senses include the five basic senses -sight, hearing, touch, smell, and taste. As these senses are stimulated in response to an external change in the environment.

For robots, exteroceptive sensing is provided by sensors and devices such as cameras, ranging sensors, radars or tactile skins. Unlike the humans, the robot's exteroceptive senses can process information from sensors placed around the robot as well as the sensors placed on the robot itself. It must be noted that these sensing systems are set up and mounted in the robot's environment, and usually require calibration routines and planning of sensor placement around the concerned volume of operation. However, due to the densely occluded nature of indoor environments and factory floors, occlusion becomes inevitable. To alleviate this problem the use of exteroceptive sensors affixed to the robot, also known as on-robot sensors is a viable option. Hence, these exteroceptive sensing can be further categorized based on the perspective of the robot as *intrinsic sensing* and *extrinsic sensing*.

As it can be verbose and confusing to refer to the aforementioned sensors with their designated terms. For convenience, the systems can be divided into two categories similar

[2] a dictionary definition of the word 'exteroceptive' Link

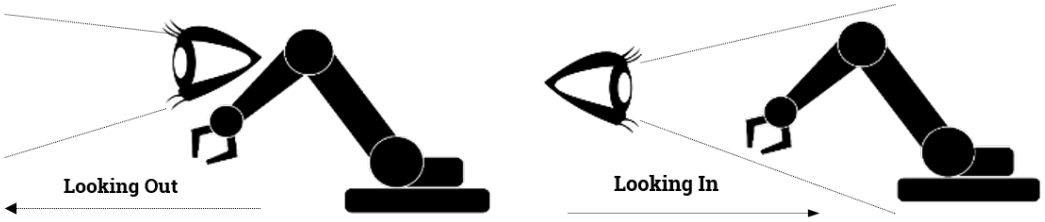


Figure 2.12: (a) A representation of intrinsic sensing i.e., looking out ('inside-out'). (b) A representation of extrinsic sensing i.e., looking in ('outside-in').

to virtual reality (VR) tracking systems. When the tracking system is completely self-contained within the VR headset it is referred to as 'inside-out' tracking. When the tracking system is completely external to the VR headset, it is referred to as 'outside-in' tracking. Similarly, when a sensing system is affixed on the robot it would be convenient to express the system as an 'inside-out' sensing system from the robot's perspective and 'outside-in' from the sensors mounted in the environment. The 'inside-out' and 'outside-in' are termed as as *intrinsic* and *extrinsic* sensing systems, respectively. A visual representation is shown in Figure 2.12.

In this research, the AWARENESS subsystem has both extrinsic and intrinsic sensors for monitoring the human-robot shared workspace.

B.1 Extrinsic Sensing The extrinsic sensors provide better contextual representation and detail of the workspace environment when compared to other sensors. A few examples of extrinsic sensors are 2D-cameras (inaccurate and inexpensive), Kinects/3D cameras (somewhat accurate and inexpensive) used in [60], 3D LiDARs (accurate and very expensive), and Motion Capture Systems (very accurate and very expensive) [55]. The drawbacks of extrinsic sensors are occlusion by other agents and objects in workspace are more susceptible to influence by other environmental factors such as light. Moreover, it requires positioning and setup for an appropriate view of the monitored workspace. The amount of computation load for deriving the information is also higher. Commonly used extrinsic sensing are motion capture systems and multiple 3D depth cameras for the generation of spatial and temporal information.

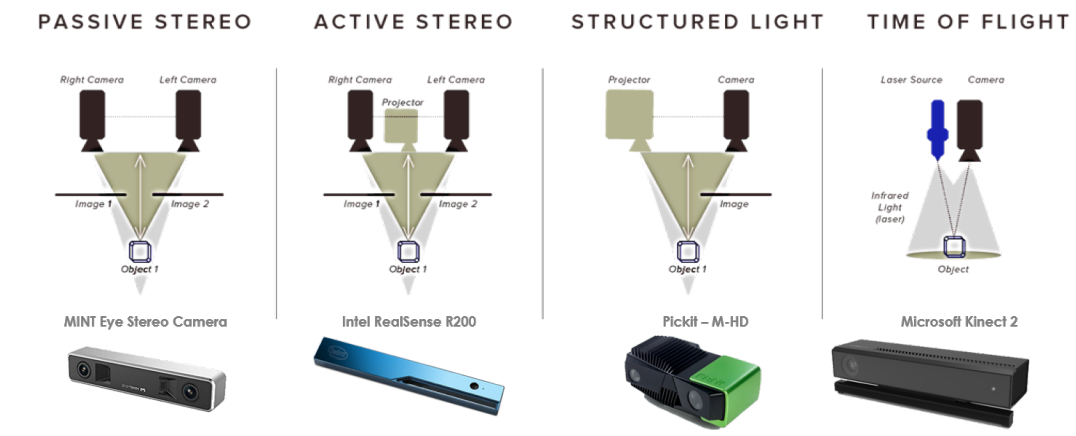
(i) 3D depth cameras can be used to generate a 3D pointcloud representation of the workspace that provides rich contextual information. There are different types of 3D

cameras that can be used for generating 3D points. They are shown in Figure 2.13(a) [61]. The passive and active stereo cameras use two or more 2D cameras, and the pixel points tracked in the images from these cameras are used to determine depth. The drawbacks are high processing requirements, low frame rate, and poor performance with texture-less surfaces. In industry, structured light and 3D Time-of-Flight cameras are generally used. The structured light cameras work by projecting a structured pattern and the distortion of the pattern is used to determine the 3D geometric shape. They are unaffected by ambient light, are fast and highly accurate, but have a shorter range compared to the 3D Time-of-Flight cameras. These are generally used for bin-picking applications in industry.

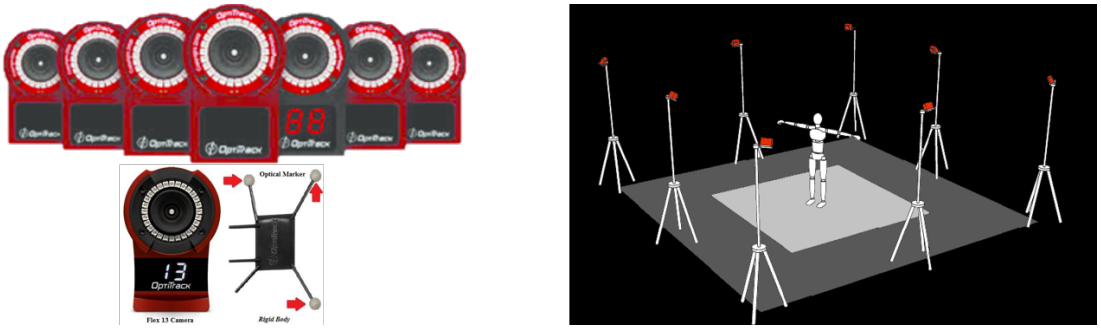
The Time-of-Flight (ToF) 3D camera (also known as continuous-wave ToF camera) illuminates the full scene with continuous wave modulated light and receives the reflected light using standard CCD or CMOS sensors. By measuring the phase shift of the received light wave, the distance between the camera and the reflecting surface is derived. These cameras have simple and compact hardware, low processing power, and have high refresh rates. The drawbacks are poor outdoor performance (under sunlight), are adversely affected by reflective materials and can have interference in the presence of other ToF cameras. As the HRC setup is generally indoors, these cameras are used in industry for human pose tracking and gesture recognition. In this research, Microsoft Kinect v2, a 3D ToF camera is explored as an option for extrinsic sensing. The research in the field of HRC have used 3D ToF cameras for the generation of 3D pointcloud of the workspace [60,62], tracking human pose [63], robot collision avoidance [18,60,64] and determining object affordances [65].

(ii) Motion capture setup is another commonly used extrinsic sensing option in HRC [24]. These systems use markers on the entities that are tracked in the workspace. The tracking performance is highly accurate for a large area and at very high speeds. For tracking, the markers should always be visible to at-least three cameras in the workspace. This setup like all extrinsic systems, suffer from occlusion related loss of tracking. This setup is commonly used for human precise human pose tracking [66,67].

In the work presented here, an OptiTrack 3D tracking system is used [68]. The OptiTrack is a motion capture system that tracks the moving objects by using Flex 13 cameras, which can reach millimeter precision for a 120 frames per second (FPS) refresh rate. The setup and the cameras used are shown in Figure 2.13(b). The real-time tracking of



(a)



(b)

Figure 2.13: (a) A set of depth cameras used for generation of 3D information of the workspace (b) A schematic of a motion capture setup used in industry. Image Source (a): DAQRI

human movement in the workspace is crucial for a human-in-the-loop system. The tracking information is useful in generating and interacting with a virtual world representation (*digital-twin* [54]) of the moving agents, which are the human, robot, and objects in the physical/real world.

B.2 Intrinsic Sensing In many indoor environments and factory floors, the robot workspace can be densely occluded with other peripheral machinery. In these scenarios, intrinsic sensing is the preferred solution. The information available from these sensors is not contextually rich and is very sparse in comparison to extrinsic sensing. The intrinsic sensors can be further categorized as follows:

(i) Tactile-based sensing has been done using a pressurized Skin [69], capacitive touch [70] and buttons on robot links or End-of-arm-tooling (EOAT).

(ii) Vision-based sensing examples are EOAT cameras for visual servoing and grasping applications.

(iii) Distance-based/Ranging sensing applications have used air curtains, 2D LiDARs [46], pressure pads, which are examples of detection from a static reference frame. The zones tracked are static and 2D in nature from the robot's perspective. For monitoring the workspace in 3D, the ranging sensors can be used by mounting them on the robot. The ranging sensors that can be used for on-robot sensing are Time-of-Flight (ToF) single unit lidars [26], laser-based distance sensors, infrared ranging sensors [20, 71] or ultrasonic sensors. These can be used for generating 3D safety zones based on the robot pose.

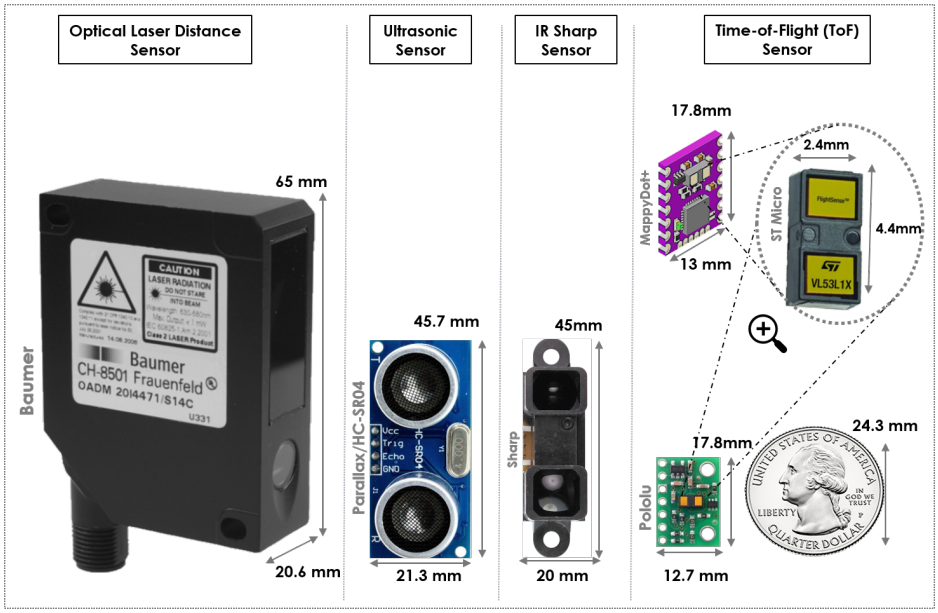
In this research, distance-based/ranging sensors are explored for intrinsic (on-robot) sensing. A comparison of commonly used ranging sensors in robotics in terms of dimensions and principles of working are shown in Figure 2.14(a) and Figure 2.14(b) respectively. For the work presented in this research, the sensor chosen for implementation of on-robot sensing is a Time-of-Flight (ToF) sensor due to its small footprint and highly directional nature of ranging [23].

2.3.1.3 Combining Physical and Virtual Awareness

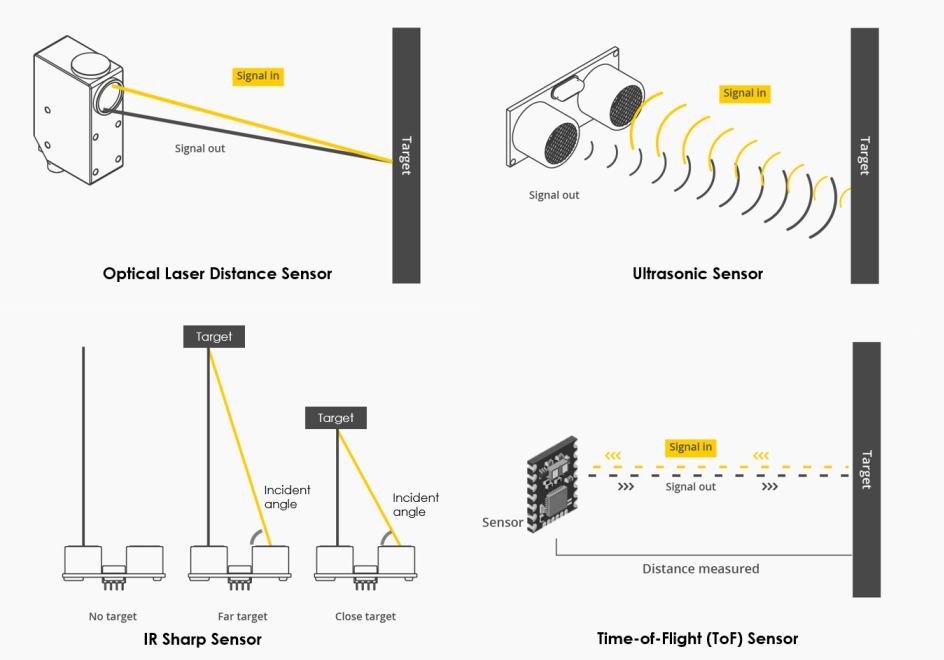
This form of Awareness entails interaction, representation, monitoring, and control of entities in the virtual world based on the physical world. The information in the virtual world can also affect a change in the physical/real world. This form of Awareness has been used in research for human-centric interaction in HRC. An outline has been shown in Figure 2.15.

A. Virtual Reality(VR) and Mixed/Augmented Reality(MR/AR)

A *virtual reality (VR)* setup was used to see the performance of an SSM configuration for an HRC task modeled in the VR [53]. Microsoft HoloLens, which are *mixed/augmented reality (MR/AR)* eyeglasses have been used in industry for training and applications in augmented manufacturing [52, 72]. A survey and applications of VR and MR/AR technologies had been presented in [73, 74].



(a)



(b)

Figure 2.14: (a) A comparison of dimensions of ranging sensors that can be used for mounting on the robot for Intrinsic sensing. (b) A simplified schematic showing the workings of the ranging sensors: optical laser distance sensor, ultrasonic sensor, infrared triangulation sensor, and time-of-flight sensor Image Source: Terabee.

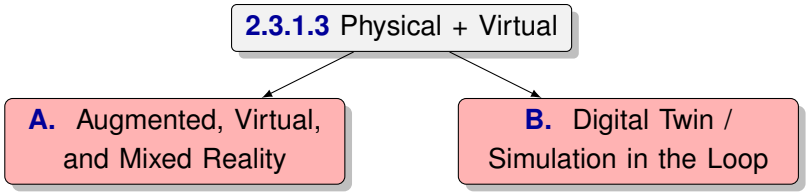


Figure 2.15: A tree diagram depicting the types of combined Physical and Virtual Awareness of HRC setups discussed in Section 2.3.1.3.

B. Digital Twin / Simulation in the Loop

A *digital-twin* can be defined as a bridge between the physical and the virtual world. In the context of human-robot collaboration, the human, the robot and the environment have digital counterparts that accurately replicate their states in the real/physical world. As represented in Figure 2.9, the AWARENESS of our conceptual system is comprised of the information presented in the physical and virtual world. A key component of the implementation of the digital twin is the use of *physics engines* [75, 76] that allows accurate virtual world representation of the physical world. Although the virtual world does not account for the uncertainties of the real world, the observable nature of the virtual representation is proven to be a powerful testing approach for algorithms and HRC experiments. Other possible applications of the digital twin are process evaluation before, during or after the execution and real-time scene assessment for revealing hidden aspects of the collaboration via visualization or another human intelligible feedback [54, 55, 77].

2.3.2 Intelligence Aspect of HRC

INTELLIGENCE is the reasoning, logic and learning that facilitates or results in a change of the HRC environment. It is responsible for actionable commands mainly to the robot and extracting information from available sensor data. In this section, the INTELLIGENCE aspect is loosely categorized based on the desired change i.e., the objective and intended application of the HRC setup. The main objectives (challenges addressed) are addressing either the safety, trust-in-automation, and productivity or all of them during HRC.

It can be seen in Figure 2.9, INTELLIGENCE is using the AWARENESS of HRC to determine robot actions and also to make the robot complaint i.e. COMPLIANCE. There is a significant research done in addressing the aforementioned challenges. This is not a comprehensive

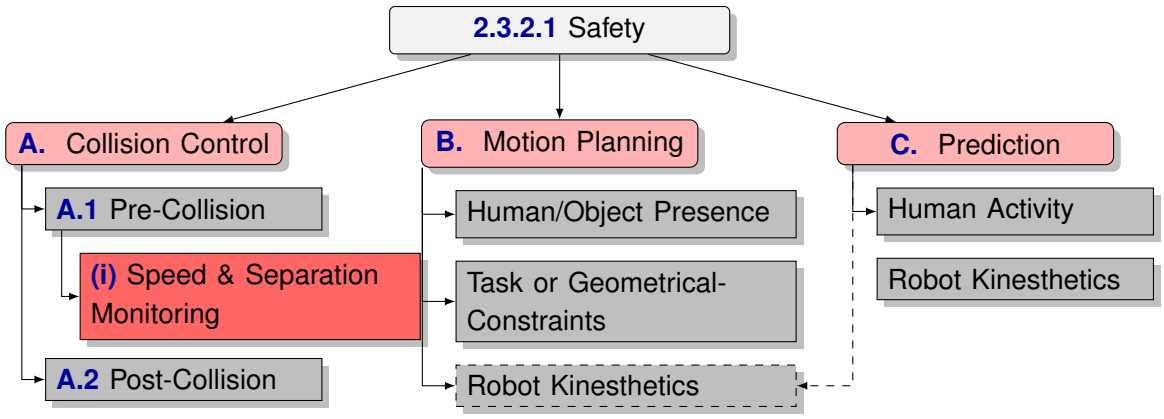


Figure 2.16: A tree depicting the outline of safety approaches in HRC discussed in Section 2.3.2.1. (The approaches that address multiple objectives are represented in dashed boxes --.)

list of all possible applications and methodologies. The commonly researched topics in HRC are considered. In this section, we focus more on the research relating to safety in HRC.

2.3.2.1 Safety

This research focuses on the intelligence needed, primarily for safety during HRC. A 2017 survey sponsored by NASA details the research done for addressing safety during human-robot interaction is provided in [18]. For safety in an industrial setup, a recent 2019 review is presented in [13].

The safety in HRC can be defined as the human physical safety for avoiding injuries due to human-robot collision, human mental safety to reduce discomfort due to robot motion, object safety when handling or grasping objects and the robot safety by avoiding collision with restricted workspace. The robot safety measures in industry outlined by the ISO standards [12,43] are guidelines for reducing risk of human injury due to human-robot collision. As described in [18], an outline for the classification of safety methodologies for HRC in industry is shown in Figure 2.16.

A. Collision Control in HRC

Collision control based safety is achieved through low-level control of the robot. This approach provides one of the simplest and fast solutions for human safety. Nonetheless,

implementation of these solutions can be complex, as they generally have time-critical constraints that require rapid execution. The control methods for safety are mainly categorized based on the robot actions pre- and post-collision. *Pre-collision control* methods are implemented before human-robot collision occurs, either by ensuring collision does not occur or by limiting robot dynamics such as speed, force, and the overall energy. In case of unpredictable or unpreventable contact occurs, *post-collision control methods* are designed to detect the collision quickly, react to minimize harm to both the human and robot, and determine robot action to recover post-collision. Post-collision control methodologies have also been used for identifying collaborative or intentional human contact, which can be used as an input for robot motion. The example of this is back-drivable motors that allow human-guided robot motion during tasks. For collaborative robots such as UR10, this mode of operation is known as 'free-drive' mode, where the robot moves based on the externally detected human force input.

A.1 Pre-collision control methods (also known as prevention methods), are intended to ensure safety during HRI by monitoring either the human, the robot, or both and modifying robot control prior to an event of collision or contact [18].

The simplest form of pre-collision control is by limiting the robot control parameters that guarantee human safety throughout the robot operation. Here, a potential risk assessment of possible hazard scenarios during the HRC task is done to determine the least safe scenario. The quantitative limits of robot pose, velocity, or exertion of forces during operation are determined as measures of risk reduction. The guidelines for these are outlined in the ISO robot safety [10, 12] and machine safety standards. An approach by [78] ensures safety by real-time tracking and limiting the overall kinetic and potential energy of the robot during robot motion. The work presented in [79] takes a unique approach of embedding injury knowledge into robot control by studying the relationships between robot mass, velocity, and impact geometry with injury. The insights from this research have been included in the new ISO technical specifications of collaborative robotics [43].

A human-robot collision can be quantified as an overlap of the human body parts with the robot in space and time. The variable for overlap in cartesian space is the distance between the human body parts and the robot. The overlap in time can be defined in terms

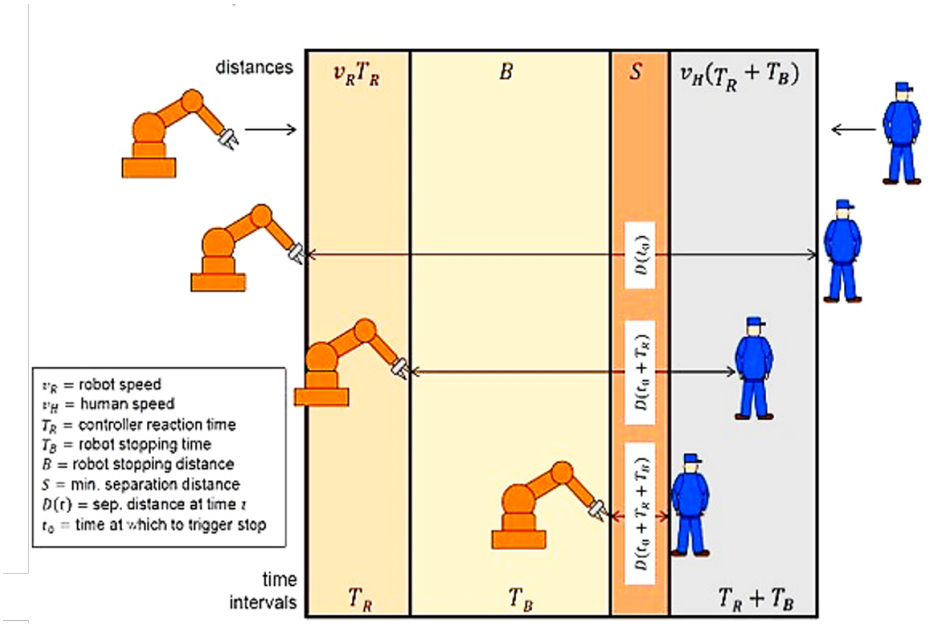


Figure 2.17: A schematic representing the process of calculating the critical/protective separation distance between the robot and the human [10].

of the velocity of the robot and human. A collision can be avoided by defining critical/protective separation distance (PSD) between human and robot following which the robot performs the appropriate remedial action to ensure human safety. The PSD between a human and robot is dependent on the velocity of the human and robot at a given time. Either robot or the human is moving at high speeds the PSD between the human and the robot will be larger. This is to take into account the processing time taken by the robot controller to process the remedial action and perform the corresponding robot movement. The schematic Figure 2.17 shows how PSD can be calculated [10]. Safety can be achieved by defining safety zones based on the PSD. The robot can slow down or stop to prevent collisions based on the intrusion of safety zones. This form of safety measure is called Speed and Separation Monitoring. In this approach, real-time accurate human pose localization is required. This is discussed further in Section (i).

Another popular approach used for pre-collision safety via robot control is by defining danger criteria and fields. The approach in [80] proposes the *potential field* approach for defining repulsive vectors based on the robot motion. This approach has been used in many research for complex safety robot behavior [60, 79, 81]. An extension of this, known as, *kinetostatic* fields was proposed in [71]. Both the above approaches are computationally

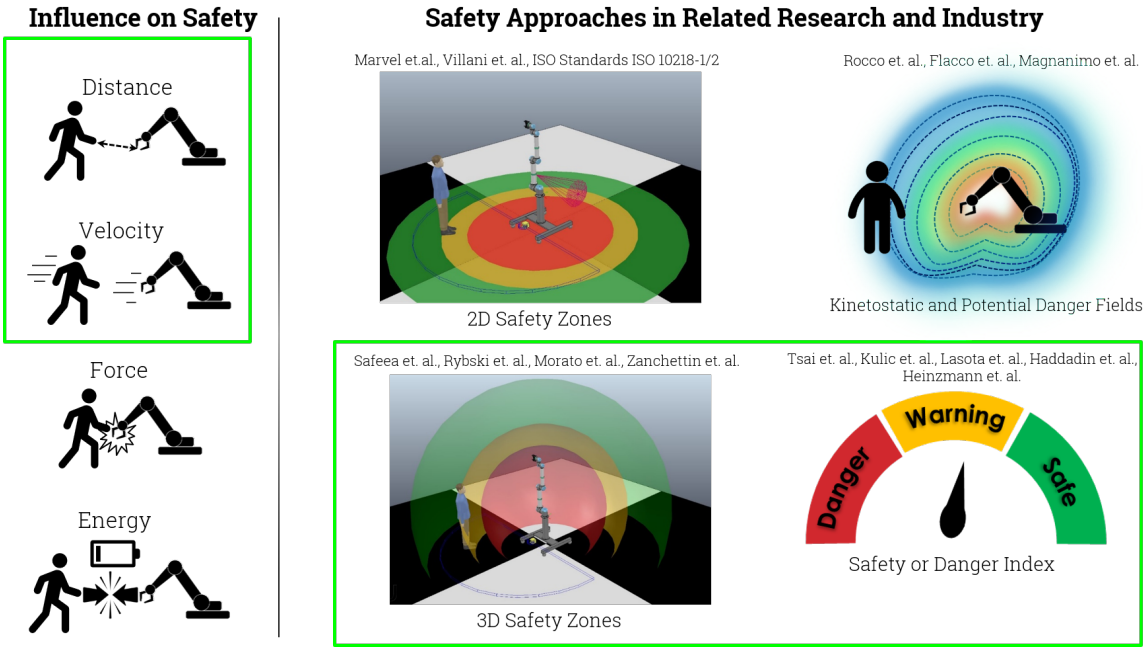


Figure 2.18: A table listing different pre-collision safety approaches used in research and industry. The highlighted box indicates the safety approaches used in this research.

intensive and require very accurate robot motion and human-robot distance measurements. Another approach is by using geometrical approximations of human and robot for measuring the minimum separation distance, as described in [55, 82].

The pre-collision safety strategies presented in research and industry are dependent on the minimum human-robot separation distance, the relative human-robot velocity, the expected force or impact of the robot, and the overall energy at the moment of impact with the human. The table shown in Figure 2.18 lists different safety approaches used in research and industry. In this research, a dynamic SSM using on-robot ToF sensor arrays is implemented based on the human-robot separation distance and velocity. The safety methodology applied is influenced by the generation of 3D safety zones and safety indexes.

There are different methodologies used for measuring human-robot separation distance and relative velocity. Figure 2.19 lists the different measurement approaches used in research and their trade-off of complexity with safety estimate and accuracy. Similarly, the complexity and the dependence on robot dynamics increases when considering force and energy for the implementation of safety. The future direction of this research extends the dynamic SSM safety setup to incorporate the robot's payload mass and shape by estimating

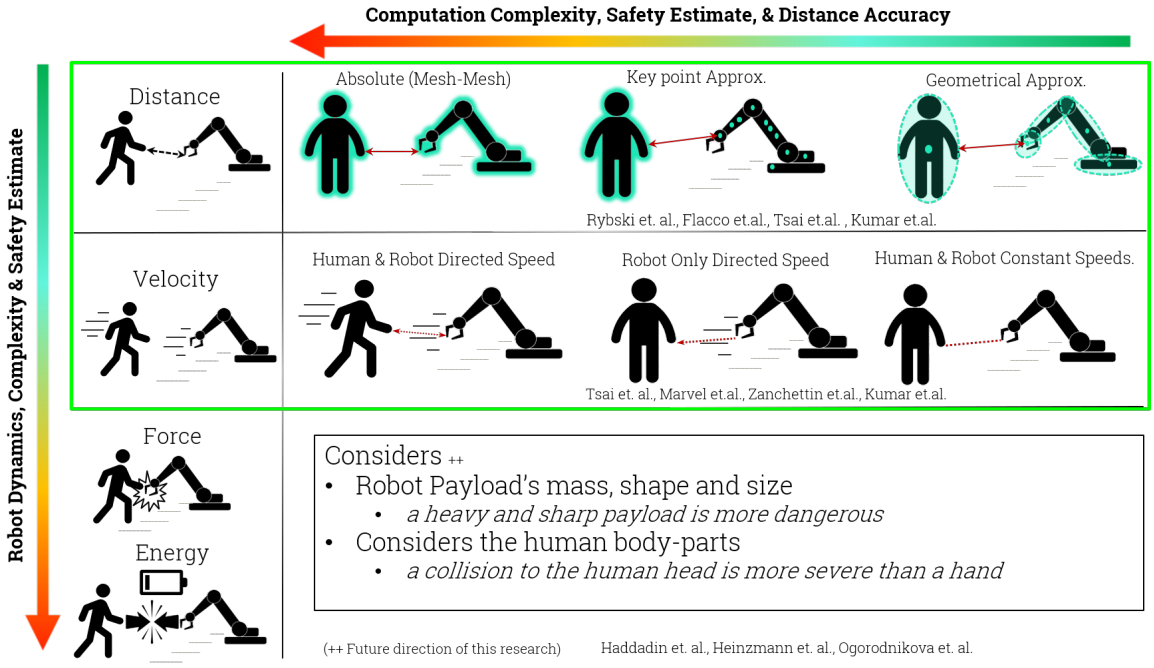


Figure 2.19: A table showing the different measurement approaches used in research and their trade-off of complexity with safety estimate and accuracy. The highlighted box indicates the measurement approaches investigated in this research.

the overall transfer of energy at the moment of human-robot collision.

The types of collaborative operations as defined in ISO 10218-1 [12] are a form of collision control methods (refer Section 2.2.3). The 'Safety Related Stops(SRS)' and the 'Speed and Separation Monitoring (SSM)' are a from of pre-collision control. Where as, 'Hand Guiding' and 'Power and Force Limiting' control can be classified as post-collision control (refer Figure 2.20).

In this research, the objective is the implementation of Speed and Separation Monitoring (SSM) using intrinsic sensing which is a form of *pre-collision* control. The related works and research have influenced the work presented in this dissertation is discussed below.

(i) Speed and Separation Monitoring In SSM, based on the intrusion in safety zones, or minimum separation distance and relative velocities between the robot and the human, the robot stops and slows its motion to avoid a collision. The conventional setups of SSM in industry use 2D scanning LiDARs to measure the distance between robot and

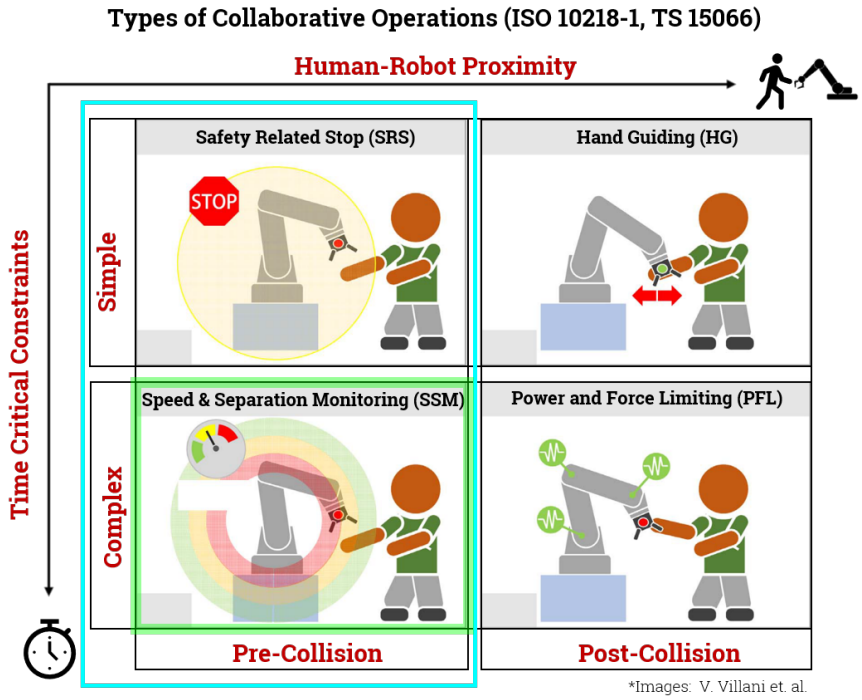


Figure 2.20: A table showing different types of collaborative operation in industry [13, 18] as pre-collision and post-collision control. The relation to the required time-critical constraints and the allowed human-robot proximity during the collaborative operation is also shown.

human. Here safety zones around the robot are demarcated using fixed distance thresholds. Such setups are inefficient in cases where the human overlaps with robot operating workspace more often, resulting in frequent robot halts, thereby negatively impacting the overall productivity. Therefore, a more flexible and dynamic implementation of safety measures that optimizes the productivity of a task while ensuring human safety is needed. Unlike the static/fixed demarcated 2D safety zones, a dynamic sensor setup that changes the safety zones based on the robot pose and motion is presented in this dissertation.

The dynamic safety zones have been implemented using extrinsic sensors in the research presented in [44, 83]. The work in [83] generates 2D dynamic safety zones based on robot pose and velocities, and projects them on the surface around the robot. The extrinsic system detects when this virtual safety zone is breached and stops the robot. By fusing sensor information from 3D stereo-vision and ranging sensors, the authors of [44] generated 3D dynamic safety zones based on human and robot motion.

Using extrinsic sensors that are positioned around the robot or on the human operator to maximize workspace area coverage has been the focus of many recent works. Recently, a 2D Lidar was used in conjunction with an IMU-based human motion tracking setup [55]. Five IMUs were attached to the upper body parts (chest and arms), whereas legs were tracked using a 2D Lidar. Their method of minimum distance calculation was derived from their previous work in [84], where QR factorization was used to find the distances between capsule approximations of human and the robot. In [60], the authors used RGB-D cameras and proposed a novel approach to compute minimum distances in depth space instead of the cartesian space and also introduced the idea of robot body approximation using few key-points.

The work presented in this dissertation relies on the contributions of the aforementioned works and also [46, 85], where the authors provided metrics for evaluation of safety and productivity of an SSM setup. In [46], two 2D lidar(s) were used to track the human position with respect to a suspended industrial manipulator. The authors suggested that there was a need to track the human pose with further precise systems such as motion capture systems. The motion capture systems and IMU-based suits are human-intrusive solutions. In industry, for SSM a non-intrusive solution is desirable. The recent FreeMove ADK product by VeoRobotics Inc. [25] implements dynamic SSM for industrial manipulators by sensor-fusion of 3D ToF cameras. This approach and the one presented in [44] are the only examples of non-intrusive human pose estimation.

All the approaches mentioned so far have exclusively used proximity or inertial based sensing modalities and approaches, extrinsic to the robot. However, in [70] the authors introduced a new type of intrinsic perspective capacitive sensor that encouraged close operation between the human and the robot. In [20], the authors assessed the placement and orientation of IR distance sensors on a robot manipulator and implemented a *kinetostatic* safety assessment algorithm [71]. They also used distance sensors for potential fields and tested the sensors placement on the robot body to examine the work space area coverage. This work was extended in [19] that described a distributed sensing system using IR distance sensors used as a safety measure. This research used IR distance (triangulation) sensors which are not very accurate and robust, and an optimal placement based on the robot link geometrical shape was required. A single IR sensor was strategically placed to maximize coverage. The *kinetostatic* fields were used to perform the safety assessment. In-

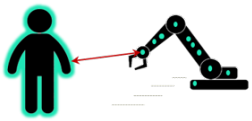
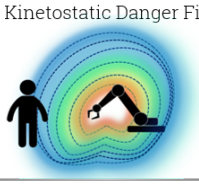
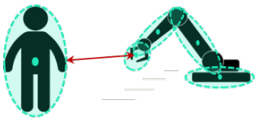
Research by	Human-Robot Distance	Safety Approach	Sensor Used	Placement on Robot
Rocco et. al.	Key point Approx. 	Kinetostatic Danger Fields 	IR Sensor 	Optimal Placements of a single IR sensor module on the robot links.
Our Approach Kumar et. al.	Geometrical Approx. 	Safety or Danger Index 	Time-of-Flight Sensor 	Placements as rings of ToF sensor arrays at the centers of the robot links.

Figure 2.21: A table showing the differences in the pre-collsion safety method presented in this research [4, 6] and in [19, 20].

fluenced by the aforementioned, the approach in this dissertation employs Time-of-Flight (ToF) sensor arrays as intrinsic sensors for implementation of dynamic SSM. A distance measurement based on ellipsoidal geometrical approximations of the robot links is used to determine placements of ToF sensor arrays modeled as rings/bands on the robot link. The Figure 2.21 tabulates the differences in the pre-collsion safety method presented in this research [4, 6] and in [19, 20].

SSM is a preferred safety measure in industry as it can be used for both collaborative and industrial robots. It has potential to be used in a variety of production scenarios as it can provide more flexibility in manufacturing for small or large robots. SSM can also be used for switching robot operation modes such as autonomous to hand guiding mode, allowing the robot to alternate between autonomous and human assisted operation [49].

A.2 Post-collision control is using control-strategies for safety through detection of and appropriate reaction to human-robot collisions [18]. The work presented in [50] uses only proprioceptive information of the robot to determine if the collision has occurred. In the case of transient contact, the reactive robot action is to move away from the point of impact. The work in [86] tested a collision detection and reaction scheme in a controlled experiment involving specialized hardware setups, including instrumented crash-test dummies. The collision detection strategy in [50] has been used to identify intentional i.e. collaborative and unintentional contact. In the case of collaborative contact, the robot can be used in *collaborative mode* for human-guided motion.

B. Motion Planning

In this approach, safety is achieved by changing the robot motion trajectory. The robot motion can be changes as a response to human or object presence. Robot path planning is used for obstacle avoidance. These obstacles may be dynamic such as human/object moving in the HRC workspace, or an object with no fixed position [60]. The robot motion can also be changed based on *geometric constraints* imposed due to restricted workspace. *Task-based constraints*, such as grasping an object based on its shape and orientation also result in change in robot pose.

C. Prediction

In this approach, safety is achieved by predicting the future actions of a human, robot, or other peripherals in the workspace. By predicting human actions, the robot can alter its behavior to provide better physical and psychological safety to the human and also manage the human's expectations. This prediction approach to safety is a higher and sophisticated level of human-robot interaction that takes human behavior into account for developing trust [18, 65, 66]. In addition to predicting human actions, the robot performs self-aware motion, also known as robot kinesthetics based motion [87] in case of a hand-over and shared tasks. For tasks such as robot catching a moving object, picking from a conveyor, requires the prediction of objects and peripherals.

In the next section, we present a few examples of research that has been implemented for increasing trust-in-automation and productivity.

2.3.2.2 Trust-in-Automation and Productivity

The 'trust' between human and the robotic automation can be achieved by managing the task-specific expectation and actions of the robot that are based on how a human operator expects the robot to behave.

" 'Automation' is technology that actively selects data, transforms information, makes decisions, or controls processes" and " 'Trust' can be defined as the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability " [42].

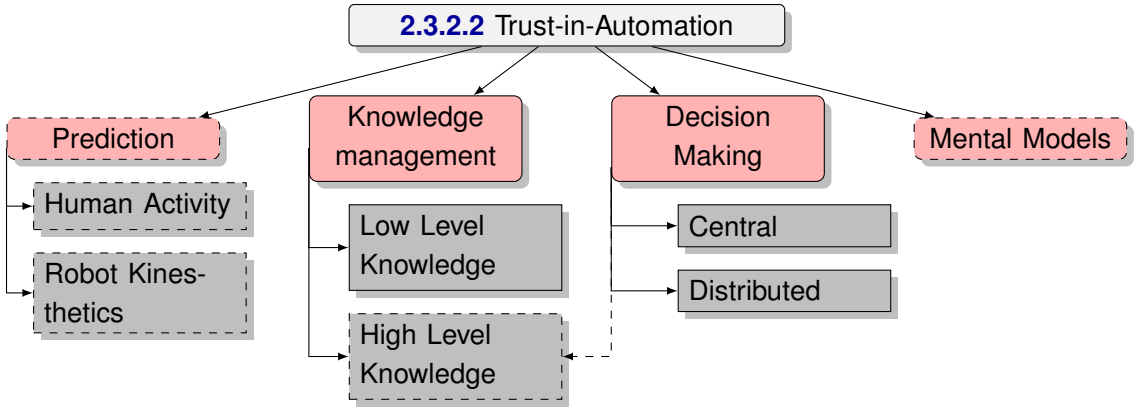


Figure 2.22: A tree depicting the various approaches for achieving trust-in-automation in HRC. (The approaches that address multiple objectives are represented in dashed boxes --.)

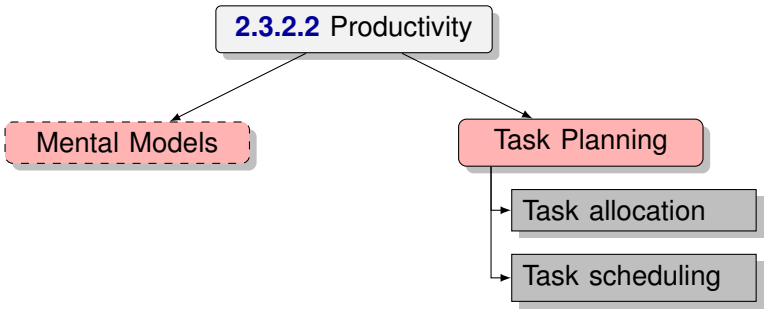


Figure 2.23: A tree depicting the various approaches for increasing productivity in HRC. (The approaches that address multiple objectives are represented in dashed boxes --.)

Hence, all approaches that help achieve reliable and accurate robotic automation will contribute in developing 'trust in automation'. Accurate *prediction* of human and robot activity can contribute to managing human expectations during HRC tasks. The reliable interpretation of sensor information and representation of *knowledge* regarding the human-robot collaboration state is also crucial for reliable decision making. A reliable *decision making* process and learned *mental models* [3] of human behavior could be used for better assisting the human during an HRC task [65, 66, 88]. A diagram depicting the various approaches for achieving Trust-in-Automation in HRC is shown in Figure 2.22.

There is a causal relationship between safety and trust-in-automation. By increasing the sense of safety for the human, the human comfort level increases for sharing the robot

[3] mental models are frameworks that individuals construct to support their predictions and understanding of the world around them

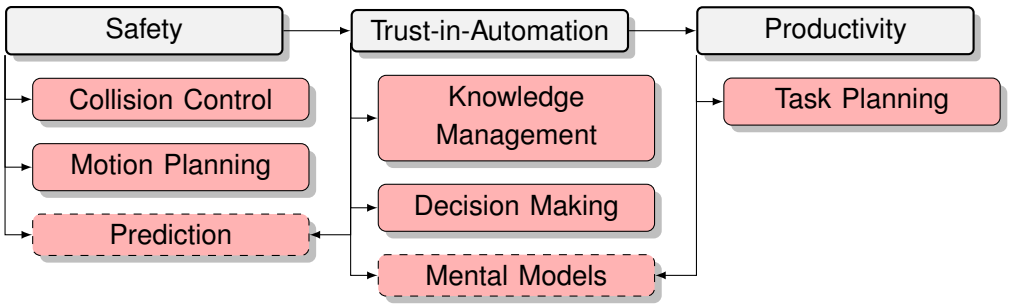


Figure 2.24: A tree diagram depicting the categorization and relation of approaches for safety, trust-in-automation and productivity in HRC. (The approaches that address multiple objectives are represented in dashed boxes --.)

workspace and collaborating with the robot. An increase in safety and trust-in-automation can also positively affect the overall productivity of the task. The relation of different approaches is depicted in Figure 2.24.

In a shared task type degree of collaboration, the productivity of the HRC task can be increased by *task allocation* and *task scheduling* [21, 66, 89, 90]. A diagram depicting the various approaches for increasing Productivity in HRC is shown in Figure 2.23.

2.3.3 Compliance Aspect of HRC

In the AWARENESS and INTELLIGENCE aspects of HRC, the human is just a dynamic object in the environment. However, COMPLIANCE of robot system tries to interpret actionable and the human mental and physical state information from human physiological signal as a form of feedback and control. This can be further used in quantifying trust in automation and evaluating human-robot interaction. There is a large body of research in the classification and interpretation of human physiological signals for both control modality and identification of human state. In order to have such a system, data collection and monitoring becomes crucial in an HRC setup.

From an HRC perspective, the COMPLIANCE aspect is about the robot managing human expectation in the form of human control over the robot and efficient communication and feedback between a human and a robot. COMPLIANCE by definition, refers to a response specifically or a submission made in reaction to a request. The request may be explicit in nature (e.g. human speech command to the robot) or implicit (e.g. interpreting human affective state) in nature. The COMPLIANCE nature of an HRC setup can be categorized as

human-to-robot and *robot-to-human*.

2.3.3.1 Human-to-Robot 'Control and Feedback'

The explicit request here is a human's direct control over the robot actions and implicit in the form of human-to-robot feedback. An outline for the COMPLIANCE approaches for human-to-robot control and feedback is shown in Figure 2.25 and discussed below.

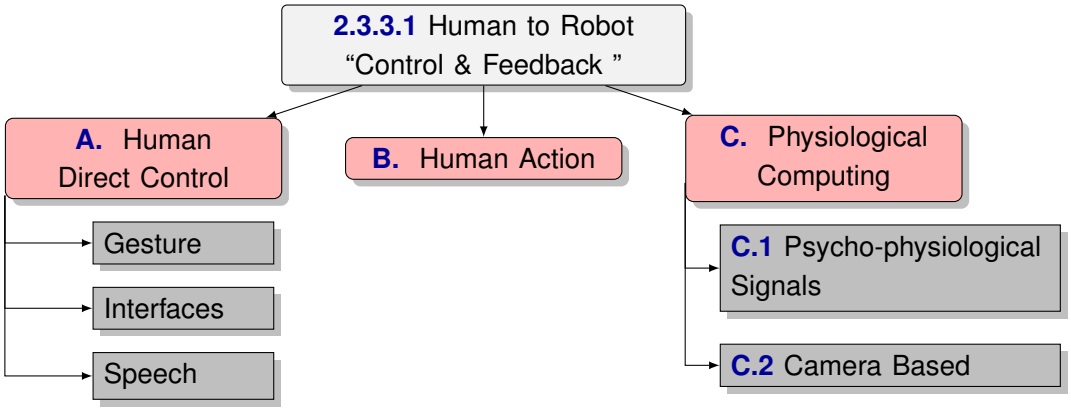


Figure 2.25: A tree depicting various approaches for achieving compliance from human to robot in HRC as discussed in Section 2.3.3.1.

A. Human Explicit/Direct Control

This is a directly commanded control from the human to the robot. The control can be in the form of hand gestures, or from direct interfaces such as joysticks and teach pendant, or in the form of speech.

B. Human Action (Implicit Feedback)

The 'implicit' feedback from the human is a derived information and interpretation by the robot. This is achieved by the HRC system having some form of 'high-level-knowledge', 'prediction', and 'mental models for human' intelligence, as referred to in Section 2.3.2 [66, 88, 89]. By predicting human actions and object affordances, the robot can anticipate human actions and assist during HRC [65].

C. Physiological Computing (Implicit Feedback)

Another form of 'implicit feedback' is by using proprioceptive information from human. This field of research is known as 'Physiological Computing', which is defined as 'the use of human physiological data as system inputs in real time—enables the creation of user-state representations so that software can respond dynamically and specifically to changes in the user's psycho-physiological state' [91, 92].

C.1 Psycho-physiological Signals The literature review in [93] highlights the use of 'Psycho-physiological' [4] methods to evaluate human response and behavior during human-robot interaction. In this research, the HRC setup would be capable of monitoring and recording the following human physiological signals. Some of the signals used are listed below:

(i) Electroencephalogram (EEG) EEG has been used for error-related potential. It has also been used to detect alpha activity, which determines attentiveness, stress, and other emotion. It can be argued that wearing an EEG cap while working can be uncomfortable. However, it must be noted that in industry, workers can wear helmets or hats. With the advent of advanced Internet-of-Things (IoT) system's wireless communication and small size factor of EEG equipment make it plausible to get such data.

(ii) Electrocardiogram (ECG) measures the heart's electrical activity. ECG can be an indicator of physical and mental stress and fatigue. In case of workers working in an industrial setup robot behavior can be adjusted based on the physical and mental state of the operator. This can increase the trust-in-automation and also avoid any injuries from work exhaustion [94].

(iii) Electromyogram (EMG) has been used as a control input for basic robot interaction in many studies. A sense of control is very important for building trust during human-robot interaction. Another example of EMG is using facial muscles to give information about sudden emotional changes or reactions. Placement of these can be in safety glasses worn by the operator [95, 96, 97].

[4] Psychophysiology is a branch of neuroscience that seeks to understand how a person's mental state and physiological responses interact to affect one another.

(iv) Electro-dermal Activity (EDA) or Skin Conductivity (SC) or Galvanic Skin Response (GSR) is a measure of skin conductivity triggered by the human central nervous system. This signal has been used in emotion recognition, in lie detectors, and as an indicator of physical and mental stress [94, 95, 98].

(v) Heart Rate (HR) and Heart Rate Variability (HRV) is a signal that can be extracted from the ECG signal. This information can give the resting or active state of a human operator.

C.2 Camera-Based implicit feedback examples are as follows:

(i) Facial Emotion Recognition cameras can be used to monitor human emotional response from the facial expressions.

(ii) Gaze Tracking is a measurement that is used to determine human visual attention and line of sight of the human.

(iii) Pupil Dilation and Fixation is a measurement of pupil diameter change. The pupil dilation can be caused by ambient light change in environment and emotional change, where as the fixation represents the level of focus and attention [58].

In this research, gaze tracking and pupil dilation are monitored during the HRC experiment. Gaze tracking is used to identify objects the human looks at in the human-robot workspace.

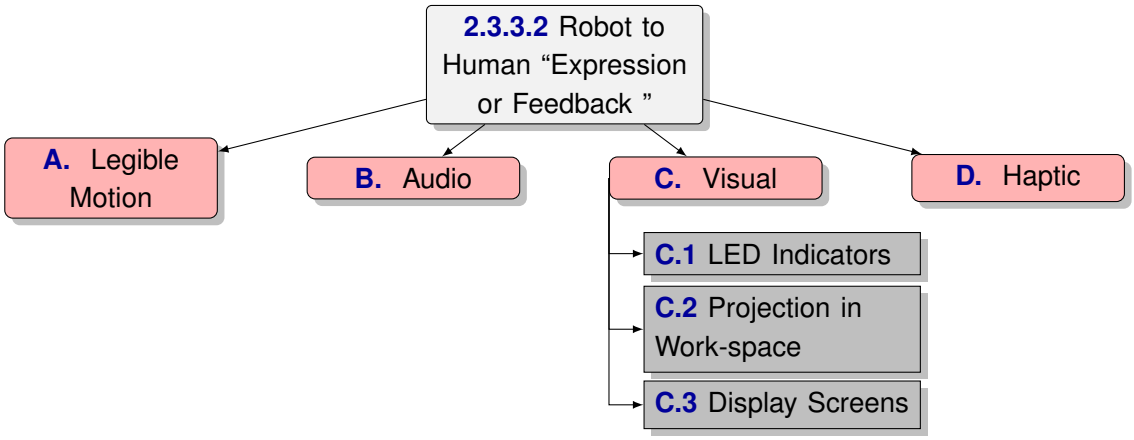


Figure 2.26: A tree depicting various approaches for achieving compliance from robot to human in HRC as discussed in Section 2.3.3.2.

2.3.3.2 Robot-to-Human 'Expression or Feedback'

The explicit response here is in the form of 'expression' from the robot to the human and implicit in the form of robot motion that is predictable by the human. An outline for the COMPLIANCE approaches for robot-to-human expression or feedback is shown in Figure 2.26 and discussed below.

A. Legible Motion (Implicit)

The robot motion being more human-like or in a way that is able to inform the human of its intentions is a form of 'implicit' feedback from the robot to the human. this type of motion is referred to as 'legible motion' [99].

A direct response from the robot-to-human is form of explicit communication or an expression. This response can be auditory, visual or haptic in nature.

B. Audio (Explicit)

The audio feedback can be speech or a sound indicative of a robot action or state. An alarm sound is a common expression used in industry for communication the initialization of robot motion and a malfunction or error in production.

C. Visual (Explicit)

Visual response of robot is commonly used in industry. The different forms of visual response used are discussed as follows:

C.1 LED Indicators are used extensively in industry to represent the state of robot operations. A work presented in [100] implements a novel signaling system by using LED RGB light strips mounted on the robot. A similar setup is implemented in this research to signal the safety stop, reduce speed, and normal robot operation modes for the SSM safety configuration. Here both the LED indicators and RGB LED strips are used.

C.2 Projection in the shared work-space This form of expression is using projectors that project image or color in an area in the shared workspace [53].

C.3 Display Screens are the most common form of expression used in industry. Many display interfaces (commonly known as Human-Machine Interface (HMI) panels) and teach pendants used in industry, display messages and responses from the robot. This form of expression can convey significantly more information to the human operator. The display screens are use most commonly as status monitors and with interactive user interfaces in industry. These display screens are also ideal for the use of Video-Plethysmography (VPG) [101, 102, 103] for determining human heart-rate and using this information as implicit feedback to monitor the human operator's physiological state.

The research conducted in [53] investigates a form of proactive and adaptive visual and auditory forms of expression from the robot.

D. Haptic (Expicit)

Haptic response is a form of stimuli that is felt by the human touch. The work presented in [104] investigates operator awareness in HRC through a wearable vibrotactile feedback haptic device. The most commonly used form of 'Haptic feedback' is in medical robotics and tele-robotics. Hand gloves for humans such as the *HaptX* [105] use opposing forces and resistance to motion to emulate weight and force experienced by the robot.

2.3.4 Tool used for HRC in this research

In this section, a few communication and simulation tools used for implementing an HRC setup in this research are briefly mentioned.

2.3.4.1 Communication Layer Tools

The communication layer is critical for sharing information between different systems during an HRC setup. A robust, efficient and timely networking is required between different aspects of the HRC setups. Listed below are some of the tools used in this research for the implementation of a distributed system to share data between, the physical, virtual and the intelligence of the HRC system-of-systems (cyber-physical system) setup.

A. ZeroMQ-An open-source Universal Messaging Library

ZeroMQ (also spelled ØMQ, 0MQ or ZMQ) [106] is a high-performance asynchronous messaging library, aimed at use in distributed or concurrent applications. It provides a message queue, but unlike message-oriented middleware, a ZeroMQ system can run without a dedicated message broker. It has a large community support and has bindings available for most of the popular programming languages such as Python, C++, Java, JavaScript and many more. The ZeroMQ messaging library provides sockets that allow concurrency and communication between multiple hosts i.e. computers, multiple processes running on the same computer and between threads of a program. This is a very useful tool, especially in research and academic settings for prototyping robust and fast distributed systems. It also allows the sockets to be configured in a variety of networking patterns such as server-client, publisher-subscriber, push-pull, router-dealer and many more. Due to this flexibility, this was used as the backbone for all communications in this research.

B. Robot Operating System (ROS)

The Robot Operating System (ROS) [107] is a flexible framework for developing robot software and interface. It has a large open-source community. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms. This is one of the most commonly used communication frameworks in the field of robotics. However, ROS [107] is currently not accepted in industry because of its lack of security, low robustness and its pseudo-real-time nature. ROS runs on a publisher-subscriber messaging pattern with a message broker. However, ROS 2.0 has recently been developed by using Open-DDS networking framework that is approved by the industry and addresses the aforementioned shortcomings of ROS. ROS 2.0 is still in an early development phase, and not completely supported by most robotic platforms. In this research, ROS is used to to monitor the system and record the generated data in the database using ROSBAG [107]. The ROSBAG allows the replay and storage of timestamped data from multiple systems in the distributed system setup.

C. Lab Stream Layer (LSL)

The Lab Stream Layer is a system for the collection of time-series data over a local network with built-in time synchronization [108]. Similar to ROS, the LSL stream is pseudo-real-time and is commonly used with biological signal collection systems such as OpenBCI, Pupil Lab, and g-tec systems. The LSL system is central to the human physiological data acquisition system for this research. All the time series data is transferred over the LSL layer to be recorded in a database for storage. Each device has an application/interface that retrieves signals in real-time and provides them as LSL streams for recording. Since LSL ensures synchronous data collection and stream, it avoids the need to perform time synchronization.

2.3.4.2 3D Simulation Tools

A 3D simulation model of the physical world is commonly used in research and industry. It provides an important method of analysis that is easily verified, communicated, and understood. Across industries and disciplines, 3D simulation modeling provides valuable solutions by giving clear insights into complex systems. There are many 3D simulation platforms and physics engines available. In research and industry, the constraints for such models are the ability to easily create and simulate the physical world, and for a *digital-twin* have a robust and fast communication layer between the physical-virtual world. Listed below are some of the 3D simulation interfaces used in this research.

A. Virtual Robotics Experimentation Platform (V-REP) a.k.a. CopelliaSim

Virtual Robotics Experimentation Platform (V-REP) also known as CopelliaSim [29, 76] is a user-friendly 3D simulation platform to create workspace and interactions of robots. This platform can be used to create virtual interactions with the simulated robot, that can affect the real robot behavior.

B. ROS Interface (RVIZ, Gazebo and MoveIT)

Robot Operating System (ROS) [107] is crucial for fast prototyping and using the vast knowledge base to implement algorithms. ROS can be used to generate and receive data

from the generated AWARENESS data of the system. As this data can be available to ROS, 3D environments such as Gazebo, RVIZ and MoveIt, and can be used to represent and process data.

C. PyBullet Interface

A physics engine simulation for performing and representing the robot dynamics and kinematics [75]. It has functionalities to integrate with various VR applications and also has deep learning and reinforcement learning platform compatibility. Thus allowing future research for mixed-reality and virtual reality interfacing during an HRC experiment. This subsystem interface can be used to virtually represent the physical world environment and use its functionalities to calculate robot dynamics, and kinematics. In this research, the PyBullet engine is used to determine the sensing performance of on-robot sensor mounted on the robot.

2.3.5 Evaluation Criteria in HRC

The evaluation of HRC can be based on a variety of different criteria. It can be based on measuring the safety of a safety configuration for HRC, the productivity and task fluency, the level of human-robot interaction, or based on human cognitive load. The evaluation criteria can differ based on the degrees and types of collaboration. The evaluation criteria can be categorized as objective and subjective measures. The objective measures are quantifiable entities during an HRC experiment. The subjective measures are feedback from human subjects that are quantified using questionnaires and interviews.

2.3.5.1 Objective Measures

A. Safety

(i) ***HR-Avg Separation Distance***: The human-robot separation distance indicates how comfortable the human was sharing the workspace near the robot.
(ii) ***Number of Conflicts/Collisions between the human and the robot***: More number of collisions would result in less trust and comfort for human.(*Note: Collision is defined here*

as the event where the robot velocity is not zero when the minimum distance between human and robot is zero.)

(iii) ***Safety Stop and Reduced Speed Events***: *Safety Stop Events*: The robot is instructed to stop i.e. the speed is set to zero. *Reduced Speed Events*: The robot is instructed to perform the task at reduced speeds.

(iv) ***Safety Metric***: The ratio of the square of the absolute minimum separation distance between Human and Robot) to the velocity of the end-effector of the robot at any given time. An HRC setup is only as safe as its least safe moment during human-robot collaboration. Thus the minimum of the safety metric calculated during the span of the experiment determines the safety during the HRC experiment. [1, 46, 85].

(v) ***HR-Intrusion Distance***: Give a minimum safety zones around the human and the robot, when the robot comes to a stop how much is the overlap or intrusion in the comfort zone of the human [85].

B. Performance

(vi) ***Velocity before and at Stop***: The velocity before and at the time of collision indicates the severity of the collision. It tells whether the robot was reactive enough to stop or reduce speed in time.

(vii) ***Human Tracking Error***: For HRC setups that require to track the human pose and actions, this criteria is critical.

(viii) ***Sensor Accuracy***: The sensing data used for perception of the environment to determine the next robot actions during HRC. If the sensors are not accurate, the robot behavior and motion would not be as desired, negatively impacting the safety, trust-in-automation and productivity.

(ix) ***Robot Jerk or Rotatum***: This measure indicates the smoothness between transitions of speed change. The lower sum of jerks/rotatum and smoother robot motion helps increase human comfort and trust in automation. Moreover, less jerks cause less wear and tear of the robot. Here we take the sum of the derivatives of the reported robot joint torques which is termed as *Rotatum*.

(x) ***Avg. Robot Stopping/Reducing Speed Time***: This measure is specific to pre-collision approaches such as SSM. This indicates the anticipatory nature of robot action to human or object presence in a shared workspace.

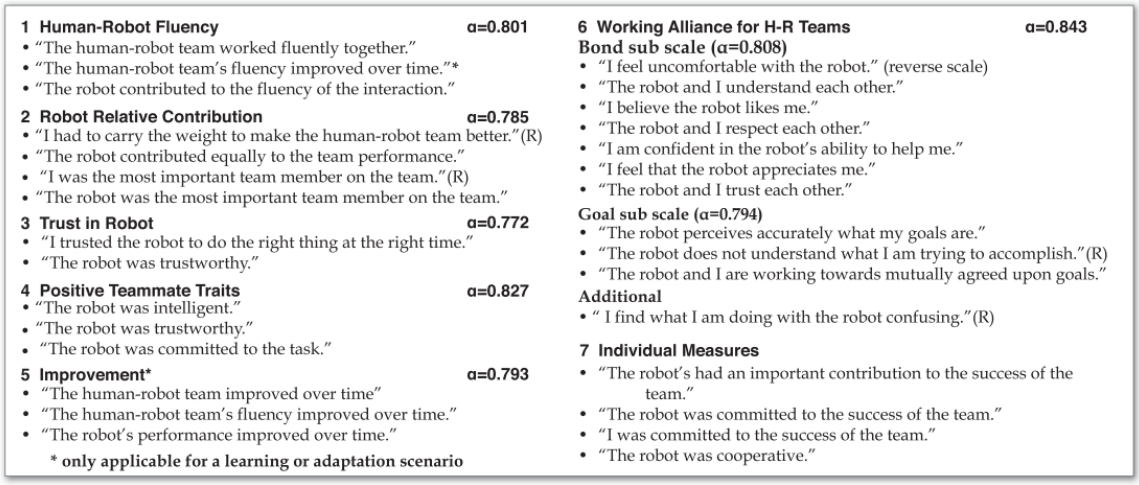
(xi) ***Avg. Robot Reaction Time***: The reaction time of the system to determine the next robot action. It is also referred to as the time interval between the robot to initiate performing the action and the time when the command to perform the action was given.
(xii) ***Task Specific Performance***: An example is tracking, picking or grasping objects of interest in the robot workspace successfully [64]. Another example is accurate interpretation of human-explicit commands. These measures are used to determine the performance of the agents that directly affects the completion of task or interaction between the agents.

C. Productivity

(xiii) ***Task Completion Time*** is the time taken to complete the task, for robot, for human or a combined task.
(xiv) ***Productivity Loss*** is defined as the ratio of the task completion time with no HRC (either done by robot or human) and the task completion time taken with the HRC setup.
(xv) ***Concurrent Human-Robot Activity (C-ACT)*** : The duration during the HRC experiment when human and robot are both actively contributing to the completion of a task.
(xvi) ***Human Idle Time (H-IDLE)*** : Human is idle because of robot lagging in supporting the human, human feels unsafe to share the workspace or due to obstruction created by the robot in the shared workspace.
(xvii) ***Robot Idle Time (R-IDLE)*** : Robot is idle because it is unable to move, cause of predefined rules to prevent human-robot collision. This reduces the productivity of the task.
(xviii) ***Number of Actions performed by the Human and Robot*** This shows the distribution of contribution of the human and robot towards the completion of the task.
(xix) ***Functional Delay (F-DEL)*** Percentage of time out of the total task time, between the end of one agent's action and the beginning of the other agent's action [21].

D. Overall-Productivity Loss vs Safety Metric

According to the formulation of safety metric as described in [85], the safety is directly proportional to the human-robot distance and inversely proportional to the robot-velocity. Whereas, the productivity loss measures the loss in productivity due to human interference



1 Human-Robot Fluency α=0.801
- "The human-robot team worked fluently together."
- "The human-robot team's fluency improved over time."*
- "The robot contributed to the fluency of the interaction."

2 Robot Relative Contribution α=0.785
- "I had to carry the weight to make the human-robot team better."(R)
- "The robot contributed equally to the team performance."
- "I was the most important team member on the team."(R)
- "The robot was the most important team member on the team."

3 Trust in Robot α=0.772
- "I trusted the robot to do the right thing at the right time."
- "The robot was trustworthy."

4 Positive Teammate Traits α=0.827
- "The robot was intelligent."
- "The robot was trustworthy."
- "The robot was committed to the task."

5 Improvement* α=0.793
- "The human-robot team improved over time"
- "The human-robot team's fluency improved over time."
- "The robot's performance improved over time."

* only applicable for a learning or adaptation scenario

6 Working Alliance for H-R Teams α=0.843

Bond sub scale (α=0.808)
- "I feel uncomfortable with the robot." (reverse scale)
- "The robot and I understand each other."
- "I believe the robot likes me."
- "The robot and I respect each other."
- "I am confident in the robot's ability to help me."
- "I feel that the robot appreciates me."
- "The robot and I trust each other."

Goal sub scale (α=0.794)
- "The robot perceives accurately what my goals are."
- "The robot does not understand what I am trying to accomplish."(R)
- "The robot and I are working towards mutually agreed upon goals."

Additional
- " I find what I am doing with the robot confusing."(R)

7 Individual Measures
- "The robot's had an important contribution to the success of the team."
- "The robot was committed to the success of the team."
- "I was committed to the success of the team."
- "The robot was cooperative."

Figure 2.27: A list of subjective criteria for Task Fluency [21].

during the HRC task. Hence, the safest safety configuration is when the robot is not moving, which also means there is no productivity, or when there is no human-robot interaction as the human is outside the workspace and the robot is moving autonomously. Hence, this measure is commonly used to determine if a safety setup is considered 'usably safe' that optimizes the safety and productivity.

Some of the aforementioned criteria have been used to evaluate human-robot collaboration systems in state of the art works like [65, 66, 67]. The objective measures for human-robot task fluency are listed and its significance described in detail in [21].

2.3.5.2 Subjective Measures

Subjective measures are generally task-specific and are based on the HRC experiment objectives. The work presented in [21] compiles a list of standard subjective criteria for task fluency that are quantified in terms of questionnaires, interviews, and observations from the human subjects during HRC experiments. This is shown in Figure 2.27. The significance term α indicates the likelihood or probability of the human-subject response for a given observation to be significant. Another aspect of evaluation is user feedback study to determine the 'trust in automation' of the human operator. The human operator sharing the robot workspace is asked to give feedback on how comfortable they were working with a robot in close proximity and whether the proposed system was able to provide information on its current state effectively. Such studies have been performed in [65] and [109]. Other

subjective criteria and questionnaires based on the human comfort levels in response to the robot's movement speed, collision avoidance, and trajectory were explored in [57,57,110].

2.4 Summary

The research presented here aims to implement Speed and Separation monitoring as the type of collaboration as outlined by the ISO Standards in ISO10218 [12]. The standards are discussed for providing the guidelines while developing the SSM safety measure and how it can be used in an industrial setting as an alternative or auxiliary to other electronic safeguards such as a 2D LiDAR scanner. For performing human subject experiments, Universal Robot's UR10 collaborative robot is used, which is a medium-sized robots with workspace similar to industrial robots [11]. Speed and Separation monitoring is applicable to both industrial and collaborative robot. SSM is a form of pre-collision safety measure that reduces the robot energy by reducing the robot speed based on human-robot separation distance and relative velocities. The objective is to compare implementation of a dynamic SSM safety measure using an extrinsic and intrinsic form of sensing. The intrinsic form of sensing options investigated are Time-of-Flight (ToF) laser ranging sensors and 2D LiDAR scanners placed at the base of the robot. The extrinsic sensors considered in this research are 3D time-of-flight depth cameras - Microsoft Kinect v2 and a motion capture system - NaturePoint OptiTrack system [24]. For human-subject experiments, a system of system framework for monitoring human bio-metrics - proprioceptive response is presented. The 3D simulation tools used in this research are V-REP [29, 76] and the physics engine PyBullet [75]. In this HRC setup a ZeroMQ messaging protocol along with Robot Operating System is used for communication and storage. For evaluation of the system, the criterion for safety, performance, and productivity is presented.

Chapter 3

Exploration of Dynamic Safety Measures in Human-Robot Collaboration

This chapter presents the exploration of components and sensors needed for implementation of a dynamic safety measure in HRC. A framework for a robot supervision system as a system of systems (a cyber-physical system) is designed highlighting the necessary subsystems and functionalities needed in a dynamic safety measure. The approaches to use simulation tools during HRC, interface for control and monitoring of the robot, and robot motions for safety are detailed. This chapter presents the results of exploration and the justification of the choices of extrinsic and intrinsic sensors used in this research for implementing a dynamic safety measure. A comparison of the advantages and disadvantages for using extrinsic sensors such as 3D depth cameras (Microsoft Kinects) and a Motion Capture system, is described. In this chapter, the ideation of the concept of using Time-of-Flight (ToF) on-robot ranging sensor arrays as a form of intrinsic sensing for a dynamic safety measure is presented.

3.1 A Framework for a Robot Supervision System for HRC

To enable a safe human-robot collaboration, a framework for a robot supervision system is designed that highlights various components and subsystems needed for achieving an adaptive control of robot motion using real-time perception. The objective is how the robot's behavior can be changed based on human actions, task execution, and changes in the collaborative workspace. The system described is a general framework inspired by the conceptual framework presented previously in Chapter 2 Section 2.3 that can be applied to most of the HRC tasks in industry. A block diagram for the framework of the robot supervision system is shown in Figure 3.1 [7]. The framework contains six subsystems: Perception, Knowledge, Virtual World, Control, Intelligence, and Interface. The following sections describe the significance of the aforementioned subsystems.

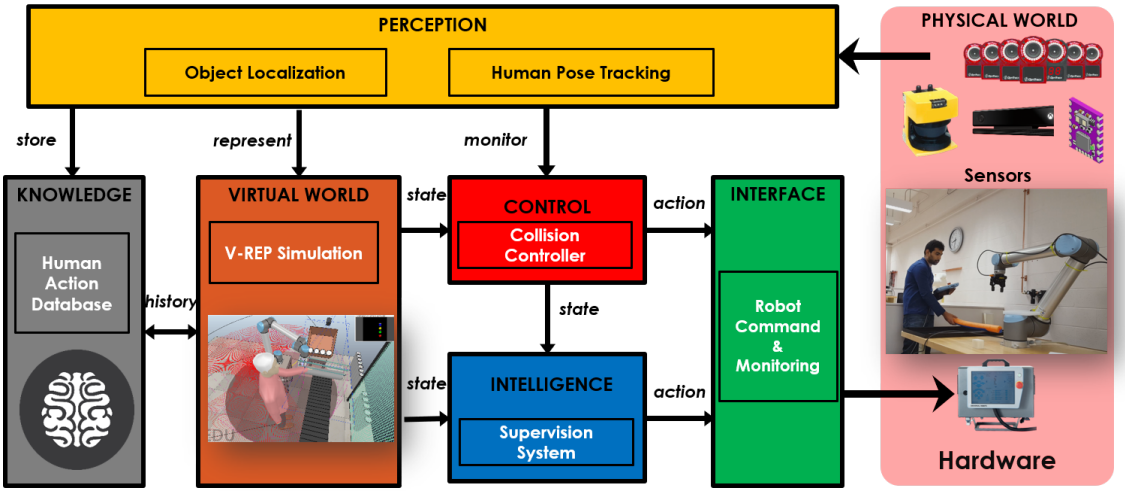


Figure 3.1: A block diagram representing the framework for a robot supervision system to adaptively control the movements of the robot by monitoring the human actions and the surrounding workspace [7].

3.1.1 Perception

The PERCEPTION subsystem is responsible for interfacing with sensors in the *physical* world that provide data regarding the workspace and the agents: robot, human, and objects of interest. This subsystem offers *awareness* for the robot supervision system by providing information to identify the state of the human-robot collaboration task/setup. For example, this subsystem can be used for human pose tracking, object localization, and human-robot distance measurements.

In this research, the PERCEPTION subsystem is used for interfacing with extrinsic sensors such as 3D depth or RGB-D (2D RGB color and 3D depth) cameras and motion capture setups, and intrinsic sensors such as on-robot ToF ranging sensors and 2D scanning LiDARs. The sensor data is essential for determining human-robot separation distance and also representing the human-robot states in a virtual world - simulation.

3.1.2 Virtual World - Simulation

This subsystem is responsible for representing the information available from the PERCEPTION into a *virtual world*, which is a simulation. There are a variety of ways simulation can be used during HRC. One of the ways is using it as a *digital-twin*. In such a setup the entire

workspace, the agents (human and robots), objects of interest, and even sensors can be represented virtually. The simulation mimics the changes in real-time based on the physical world by using the information from the PERCEPTION subsystem regarding the human, object, and robot motions. Simulation allows the representation of a state of the entire HRC setup by combining information with respect to a common reference frame. This representation can then be used to generate synthetic data, such as human-robot relative speeds, separation distance, or prediction of human-robot actions and object affordances.

Another way is using the simulation to simulate human-robot collaboration scenarios and generate data to determine and validate robot behavior. This is commonly used in industry to define the robot motion and analyze production setups before deployment. This use-case can be further extended to replaying human actions in the workspace and recreating human-robot collaboration scenarios in simulation. It can be used for testing robot behavior for a known human action or as an input for learning human actions.

Figure 3.2 shows the robot work-space used in this research modeled in a virtual-world using V-REP (also known as CopelliaSim [29, 76]) is shown. In this research, to achieve safety during human-robot collaboration requires information regarding the human-robot separation distance, human positions, and relative speeds. A *digital-twin* setup using V-REP is implemented where a human-avatar is moved according to the human-pose tracking data obtained using an extrinsic form of sensing, such as the motion capture system. In addition, 3D depth cameras are simulated to validate the sensor placement and coverage of the workspace. The V-REP simulator has been used to emulate intrinsic sensors such as the on-robot ranging sensors and 2D LiDARs.

Other physics engine based simulations such as PyBullet, Gazebo, and RVIZ have been used for modeling and testing the behavior of on-robot ranging sensors by representing the data provided by these sensors with the robot.

The on-robot ToF ranging sensor setup in this research for implementing a dynamic safety measure is achieved by mounting the sensors on the major links of an industrial robotic arm (explained further in Section 3.6). The sensor setup is simulated using V-REP 3D simulation software. In order to analyze and validate collision detection strategies using this sensor setup, a standard pick and place task with a human-avatar is simulated. The ToF sensor setup is modeled according to the characteristics of the ToF laser range sensor. The industrial robot used is a 6 degree-of-freedom (DOF) UR10 collaborative robot. In order

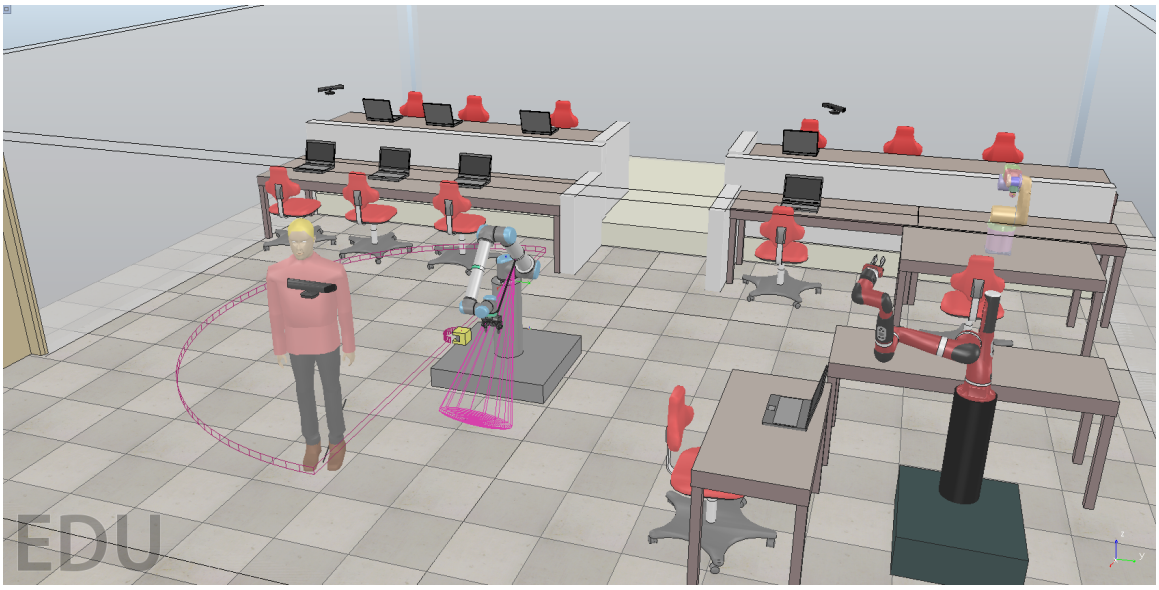


Figure 3.2: This shows the virtual world representation of the robot workspace modeled in simulation using V-REP (a.k.a. CopelliaSim). Video Link

to ensure the reaction time and fidelity of the robot movement, the V-REP simulated robot is synchronized to the real world by connecting to the actual robot task programmed on the UR10 robot controller interface called the UR Polyscope (Universal Robots Graphical Programming Environment). Thus, the UR10 robot exists in the physical and the digital-twin/simulation concurrently.

3.1.3 Knowledge

This subsystem acts as a storage and provides an advantage to recreate the simulation of human-robot tasks by recording the sensor data during the task. This subsystem can also be used to generate simulated data to test cases of human-robot collaboration. In the case of learning and predicting existing data for training and generation of mental models of robot behavior. This subsystem is responsible for collecting or providing the human, robot, and environment information to other subsystems of the framework. In this research, a tool called ROSBAG is used for time-stamped recording and playback of sensor data for simulation.

3.1.4 Control

The constraint of robot motion during a task can be based on human/object presence or geometrical and task constraints. The CONTROL subsystem is responsible for ensuring the robot motion is performed within the pre-defined constraints. There are two forms of control scenarios *pre-Collision* and *post-Collision*, where *collision* is defined as any action taken by the robot that deviates from the nominal/normal action (refer Section 2.3.2.1 A.). This defines a set of rules for robot motion while performing the task. The CONTROL subsystem determines the action to be taken in case the robot-motion is about to violate the rules (a pre-collision scenario), and the robot action taken to recover from a collision (a post-collision scenario).

In this research, the pre-collision control implemented is based on Speed and Separation Monitoring, which is reducing the energy of the robot motion (speed or force) based on the human-robot separation distance. In order to ensure the safety of the human, the robot itself, and other peripheral entities, the objective is avoiding a collision.

3.1.4.1 Robot Motion in SSM

Similar to the work presented in [85, 111], the controller outlines the behavior of the robot based on the distance between a human and a robot or based on safety zones in the workspace. It demarcates three states of the environment for HRI defined as follows:
(1) **Safe State**: the distance between the operator/object and the robot is larger than the pre-imposed boundary threshold. This distance is termed as *the critical/protective separation distance* (also referred to as the minimum allowed separation distance). The robot motion is nominal/normal.
(2) **Warning State**: where there is a possibility of a collision, but the relative distance allows the execution of avoidance or generation/update of trajectory. Avoidance means that the robot deviates from its current trajectory to avoid collision and recovers by returning back to its current trajectory. Whereas an update of trajectory happens if a new path needs to be planned to avoid the collision. The generation or update of trajectory is not a trivial task and requires consideration of many variables and constraints. Thus, the robot motion planning is performed by the INTELLIGENCE subsystem.

(3) **Danger State**: where the distance is shorter than the one associated with a high collision risk. This distance is dependent on the breaking and response capability of the robot hardware [12, 46]. Here the robot comes to a safe stop (refer Section 2.1). The action following a robot stop is to recover its motion when the human is out of the danger area.

The output of the CONTROL subsystem will interact with all the subsystems. The output of the collision controller is the generation of robot actions based on the information interpreted from the PERCEPTION subsystem, represented in the VIRTUAL WORLD and directly controls the robot movements using the INTERFACE subsystem. The CONTROL subsystem converts robot actions into robot commands as interpreted by the robot. The following section describes the INTERFACE subsystem comprising of interfacing with the robot and peripherals in the HRC setup.

3.1.5 Interface

The robot motion commands are the elementary operations of a robot action. This system is a simple interface for monitoring and executing robot motion commands in real-time for the specified robot (here Universal Robot UR10). The actions and commands determined by other subsystems for the robot and other peripherals such as LED indicators are translated and executed by the INTERFACE subsystem. Another essential function of this subsystem is the maintenance of a communication layer between various subsystems, the robot, and the peripherals. In this research, this functionality is implemented using ROS and ZeroMQ (refer Section 2.3.4.1).

3.1.6 Intelligence

This INTELLIGENCE subsystem implements the logic and reasoning for determining the robot actions and behavior based on the interpreted state of the environment and the agents (refer Section 2.3.2). The system will analyze the state of the environment and determine what action(s) the robot should take. The INTELLIGENCE subsystem in the framework is the learning model for the robot to determine the best policy to choose what action to choose based on the state of the environment to achieve the objective of the HRC task. The implementation of this subsystem requires the prediction of human actions in the robot collaborative

workspace. For the robot to plan to avoid any collision or impedance of its path, it needs to anticipate the region the operator will occupy. This robot planning requires observations and prediction models of human operator activity and motion, respectively.

For the implementation of a dynamic safety measure, the critical/protective separation distance (PSD) between a human and robot or a dynamically changing safety zone is dependent on the velocity of the human and robot at a given time (refer Section 2.3.2.1 A.1 (i)). The PSD takes into account the processing time taken by the robot controller to process the remedial action and perform the corresponding robot movement. If the robot or the human is moving at high speeds, the critical separation distance between the human and the robot or the safety zone area will be larger. In this research, for implementation of dynamic SSM as the dynamic safety measure, one of the roles of this subsystem is to determine the critical separation distance thresholds for the CONTROL subsystem to determine the state of the robot by calculating human-robot relative speeds based on the human and robot motion.

Simply put, the purpose of the INTELLIGENCE subsystem is to supervise the robot actions based on the perceived state of the environment. In this system, the dynamic agents monitored are the human and the robot, and the agent controlled is only the robot. The human activity influences the state of the environment and thereby the robot behavior.

Summary

The framework presents an overview of the components needed in the implementation of an HRC setup. This framework was used as an outline for the implementation of dynamic Speed and Separation monitoring HRC setup and experiments. As a digression, the following section presents a parallel perspective on the design of this framework with a well-known *Skill*, *Rule*, and *Knowledge* (SRK) classification of human behavior and decision making [22].

3.2 A parallel perspective drawn to the SRK Model

The ultimate objective of human-robot collaboration is enhancing the capabilities and intelligence of the robot to match the human intelligence and help transform them from ‘tools

to teammates'. A common theme and inspiration in the field of HRI is mimicking human-human teams and deriving mental models for the robot [65,66]. Hence, designing a system derived from the human behavior models is a logical choice. Here, an explanation of the SRK human behavior model with an example and its comparison to the framework is presented.

A well known human behavior model was proposed by Rasmussen [22] known as the 'Skill, Rule and Knowledge (SRK) human performance model.' A parallel between the SRK model and the framework to control robot action is drawn and explained further. According to Rasmussen's SRK model, a *skill-based behavior* of a human represents sensory-motor performance during an activity that does not require conscious control and can be done without putting much thought into it. During this the human also performs constant attention checks, to perceive visual and audio information to be aware of his/her surroundings.

For understanding, an example of a human driving a car on the road is taken. For a human who is an experienced driver, the act of driving and controlling the car can be considered as *skill-based behavior*. During this, the human also performs constant attention checks to perceive visual and audio information to be aware of his/her surroundings. A *rule-based behavior* is defined as a sequence of actions in a familiar situation or environment that are governed by a *stored* rule. In the case of a human driving, the human stops the car on a red traffic signal or takes a turn while driving on a familiar route to reach a destination. These actions can be classified as a *rule-based behavior*. Whereas in unfamiliar situations, where the environment is not known, the level of performance is goal-controlled and knowledge-based. This would be the case when a human is driving to a destination whose route is not known, and he/she has to follow the signs on the road. Another example of *knowledge-based behavior* is while driving there is an obstacle on the lane, and the next lane is occupied by another car going in the same direction, what is the course of action the driver should take. Here the goal of the human is to avoid the obstacle and not stop the car. Hence this would require more analysis of the situation by the human before taking any action. A simplified illustration of the three levels of human performance as a result of SRK behavior is shown in Figure 3.3(b).

The robot performance according to the designed framework is represented as an SRK model, as shown in Figure 3.3(a). A skill-based behavior in the robot supervision system

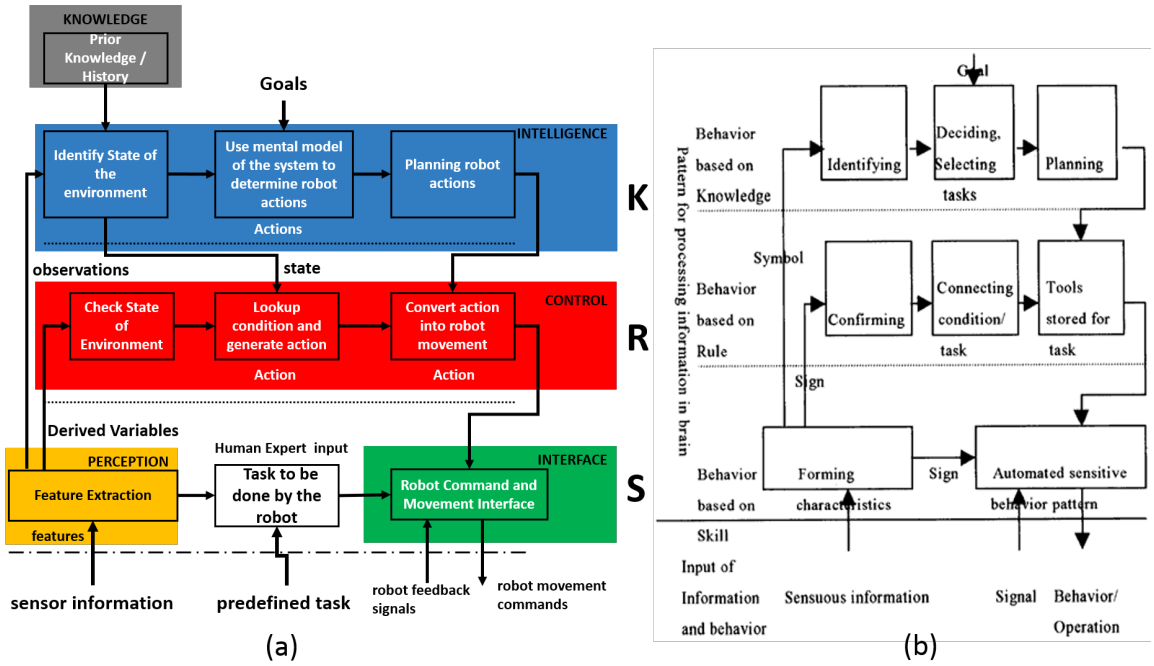


Figure 3.3: (a) A robot behavior model represented as a SRK based human performance model [7] (b) SRK based human performance model by Rasmussen [22].

contains the generation of information, representing the state of the environment and the movement of the robot. The robot autonomously performing a given predefined task by a human expert can be considered as a skill-based behavior. The extraction of features from the data from various sensors that represent the state of the environment can also be considered as part of the skill-based behavior. At the skill-based level, the PERCEPTION and INTERFACE subsystems of the framework provide the basic functionalities of robot motion for a given task.

The CONTROL subsystem is responsible for checking the distance between the human body and the robot using the perceived features and, based on the separation, determine three states: Danger, Warning and Safe states. These states are determined based on predefined threshold values and results in robot actions taken accordingly. This form of behavior by the robot can be classified as a rule-based behavior. This holds true even for the conventional measures of human safety in industry while working with a collaborative robot.

In a conventional setup with a UR10 robot, where the distance between the robot and human is measured using a 2D range scanner (Speed and Separation Monitoring [85]),

safety zones around the robot are demarcated using distance thresholds that define safety zones, warning zones and danger zones. In such cases, the feedback is provided by the tri-colored LED indicator representative of the robot state: green when the robot is moving normally, yellow when moving at reduced speeds and red when stopped. This form of feedback is important for achieving trust-in-automation between the human and the robot. Similarly, based on the information provided by the CONTROL subsystem, visual and audio feedback can be provided to the human sharing the workspace. The case where the robot needs to update its trajectory of motion or predict human actions, identification of the state of the environment is required. Based on the state, actions of the robot are determined and planned intending to complete the task and also ensuring human safety. The INTELLIGENCE subsystem of the framework is responsible for this form of robot's knowledge-based behavior. The control of speeds and generation of visual and audio feedback are done using the INTERFACE subsystem.

3.3 Sensors for Perception

The objective of this research is to explore the viability of on-robot ranging sensors for implementing a dynamic SSM safety setup, which is a pre-collision safety measure. This section details the choices and experiments to determine the sensors used for the PERCEPTION subsystem for the implementation of dynamic SSM. The following sections explore the extrinsic and intrinsic sensing options that can be used for implementing dynamic SSM.

3.3.1 Extrinsic Sensors

For implementing a dynamic SSM safety setup, spatial and temporal information such as the human-robot separation distance and their velocities are needed. From an extrinsic perspective, this requires representation of the environment and the moving agents (human and robot) in 3D space with respect to a fixed reference frame of the monitored workspace in real-time. Based on the literature, the two most commonly used extrinsic perspective sensors are 3D depth cameras [44, 47, 60, 64] and Motion Capture [53, 66, 83] (also known as mocap) systems. First is as a comparison to the intrinsic on-robot ranging sensor-based dynamic SSM implementation, and second is to use it as a ground truth to analyze the

measurement accuracy of these sensors.

The contextual information represented by extrinsic sensing is more detailed in comparison to the limited sensing of sensors from the robot perspective. This information can be used to model the workspace, track or predict human pose/actions, manipulate objects, and determine robot behavior. However, it is to be noted that for implementation of dynamic SSM, only the human-robot distance and velocity information is required. If human action prediction or determining the task progress or any higher-level knowledge needs to be derived, using intrinsic sensors would be less feasible and complex. In extrinsic monitoring, the vision sensors are stationary monitoring a fixed workspace.

On the other hand, sensors mounted on the robot have limited monitoring of the workspace. The field of view (FOV) of the on-robot sensors changes with the robot motion or is fixed from the robot's perspective. The fixed reference frame for the extrinsic sensors is the world or universal frame that is static and is chosen based on the available sensor data. As for intrinsic sensors, the sensor information is represented with respect to the reference frame of the robot link or the base of the robot.

In this research, the extrinsic sensors explored for the implementation of dynamic SSM are:

(1) **3D Depth Cameras** - Multiple 3D depth cameras are placed around the robot to monitor the workspace. The depth information is used to create a 3D point-cloud of the workspace. Based on the occupancy of these point-clouds belonging to the human and the robot, the minimum separation distance is calculated. The sensors used for this research is a Microsoft Kinect v2. This a Time-of-Flight based 3D depth camera, that provides an accurate depth map in comparison to other forms of 3D depth measurements (refer Section 2.3.1.2 B.1 (i)).

(2) **Motion Capture System** - This setup uses cameras to track optical markers placed on the human, robot, or objects of interest to track their position in 3D space. Instead of a point cloud, the markers are localized from a given reference-frame. This form of 3D localization is highly accurate and can be used as ground-truth (refer Section 2.3.1.2 B.1 (ii)). A Nature-point OptiTrack motion-capture system is used in this research [68].

3.3.2 Intrinsic Sensors

The intrinsic sensors are used for monitoring the workspace from the robot's perspective. Thus, intrinsic distance-based sensors do not provide detailed information of the workspace and human actions, but only a distance measurement of an entity in the sensing zone. Unlike vision sensors, these do not provide rich contextual information and are not a solution when the complexity of processing information increases with the increase in the degree of collaboration. For collaborative operations that solely require the spatial information of the robot's surroundings, intrinsic sensors are the preferred choice. For a dynamic SSM implementation, the human-robot separation distance is measured using distance-based/ranging sensors.

In industry, an SSM safety configuration is implemented using a 2D LiDAR to determine the human position in a 2D plane (see Figure 3.4). In the robot cell, a 2D LiDAR is mounted closer to the floor and near the base of the robot with the 2D LiDAR scanning zone overlapping the collaborative workspace [46, 55]. The measurements from a 2D LiDAR is from the sensor reference and human legs or body in a 2D plane. This is not an accurate measurement as the separation distance could be smaller than the measured distance due to human and robot pose. Thus, an on-robot sensing approach by placing distance sensors on the robot would allow distance measurements based on the robot pose.

Hence, in this research, ranging sensors that can be mounted on the robot are explored. The constraints for placing these sensors on the robot are:

(a) **Accuracy** - The sensors should have accurate and robust measurements to ensure safety using SSM during HRC.

(b) **High Frequency of Measurement** - As the sensors will move with the robot, the ranging should be fast enough to perform measurements of moving human/objects from a moving reference frame.

(c) **Long Range** - The sensors should be able to measure farther (at least equal or greater than the maximum reach of the robot) to anticipate the motion of the human/objects.

(d) **Eye Safety** - Most of the accurate ranging sensors are light-based. For HRC, the sensors are required to be eye-safe for humans to share the workspace safely.

(e) **Small Form Factor and Minimal Weight** - As these sensors are placed on the robot links, they should be lightweight not to add additional load to the robot. Moreover,

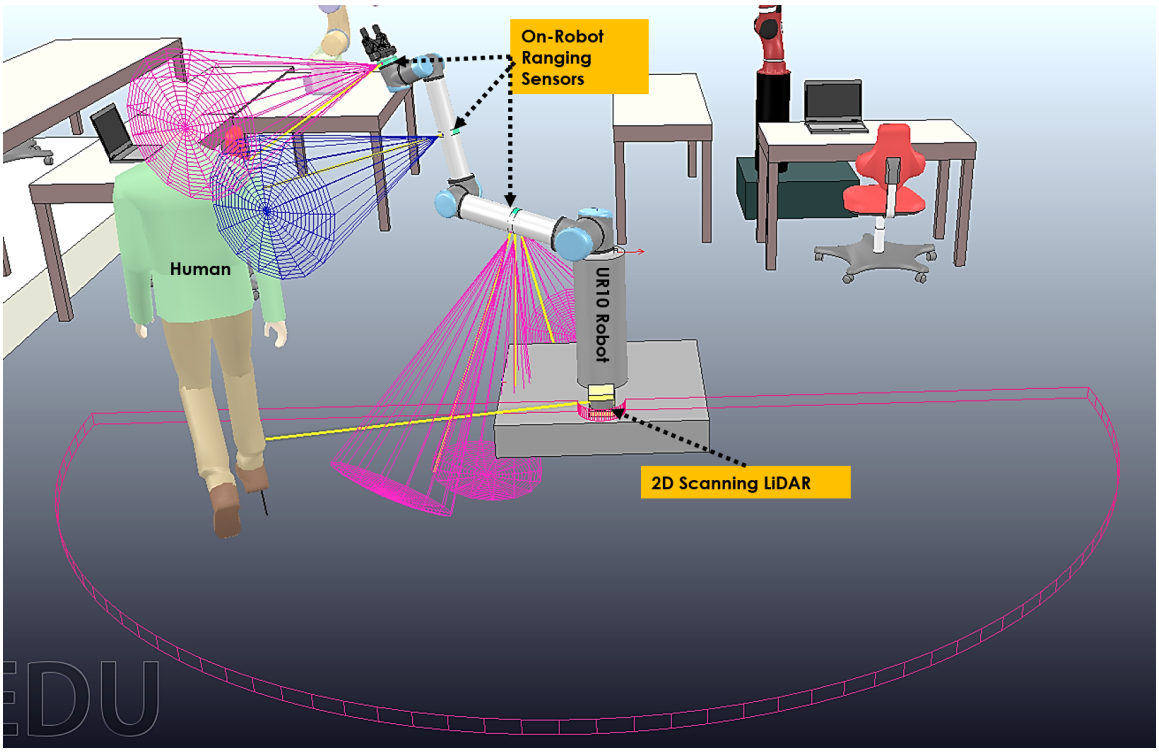


Figure 3.4: A simulated setup showing the intrinsic sensor modalities - a 2D scanning LiDAR and on-robot Time-of-Flight (ToF) laser ranging sensors explored in this work.

a small form factor is desirable as multiple sensors are required to be placed on the robot for maximizing coverage and reducing the clutter.

(f) **Multiple Sensors** - For easier and less cluttered cable management, it is beneficial if the ranging sensors can be daisy-chained or connected in series.

The commonly known ranging sensors used in robotics are ultrasonic ping sensors, infrared (IR) triangulation sensors, laser-based distance sensors, and Time-of-Flight (ToF) sensors. A comparison on the working and size is discussed in Section 2.3.1.2 B.2 (iii). Comparison based on the constraints for placing these sensors on the robot are shown in Figure 3.5 [23].

IR based sensors have been used in [19, 20] as on-robot ranging sensors as they have a small form factor and minimal weight. However, this work investigated the optimal placement of these sensors on robot links to maximize coverage for kinetostatic safety assessment. IR based sensors lack range, accuracy and are not robust as they are prone to interference from ambient light. They are also unable to detect transparent objects.

	Ultrasound	Infrared triangulation	Laser	Single Unit Time-of-Flight (ToF)
High reading frequency			•	•
Long range			•	•
Minimal weight	•	•		•
Small form factor	•	•		•
Eye safety	•	•	Class 1 laser only	•
Use with multiple sensors				•

Figure 3.5: A comparison of ranging sensors: ultrasound, infrared, optical sensor and time-of-flight that can be used for on-robot intrinsic sensing [23]

Ultrasonic sensors are sound-based distance sensors akin to sonars. These sensors are not long range and inaccurate under fast motion. Both ultrasonic and IR sensors have a wide field of view for a distance measurement, which makes them more prone to cross-talk. As multiple sensors are to be used, errors in measurement due to cross-talk are likely to happen.

Laser-based distance sensors and Time-of-Flight laser ranging sensors are light-based distance/ranging sensors. Laser-based distance sensors are highly accurate, robust, highly directional (smaller FOV, less cross-talk), and have a very high frequency of measurement. However, they do not have a small footprint and are relatively heavy in comparison to infrared and ultrasonic sensors (refer Figure 2.14(a)). The main drawback for these sensors, is that the light used is a laser and is not 'eye-safe'. Hence, this is not a viable option for using as an on-robot ranging sensor.

The Time-of-Flight (ToF) laser ranging sensor, also known as a single unit lidar, is a new technology that has gained a lot of importance. Even the 3D depth cameras such as Microsoft Kinect v2, use Time-of-Flight concept for determining the depths. This is because ToF sensors work at a higher wavelength of IR and use high-frequency pulses and phase difference to determine the distance. Unlike the laser-based distance sensors, the ToF sensors are eye-safe and can be used safely around humans. These sensors are more accurate, robust, and less prone to cross-talk because of their directional nature and

work under motion. Due to higher wavelength, they can detect most of the transparent objects [23, 26]. The sensors have a smaller footprint in comparison to all the above-mentioned sensors. Recent ToF based technologies provide a compact sensing package and faster digital communication protocols such as SPI and I2C that allow reading from multiple sensors connected in series and design of custom sensor arrays. Hence, due to these advantages the single unit ToF laser ranging sensors are used as an intrinsic form of perception that can be placed on the robots for monitoring the environment. The ideation and reasoning for use is explained further in the this chapter.

Another ranging sensor that is used with automobiles is a millimeter-wave radar [112]. Radars are great choice in terms of accuracy. Current sensing technology is being developed for radars that have a smaller form factor and also be able to measure moving objects from a fast moving reference frame. The advantage the radars can provide is multiple measurements in a wider field of view, similar to a 3D depth camera and can determine the measured entity is a human or not.

Summary

To summarize, the intrinsic sensors explored for the implementation of SSM are:

- **2D Scanning LiDAR** is currently the conventional approach for Speed and Separation Monitoring. This is a static approach where the safety zones are fixed and do not change according to the robot pose. This is used as a comparison for static and dynamic SSM.
- **On-Robot Ranging Sensors** are the less-explored option. These are sensors that can be mounted on the robot and can be used to measure the separation distance from the robot perspective. In this research, an array of single unit ToF laser ranging sensors are used to implement dynamic SSM.

There are two goals of implementing extrinsic perception-based dynamic SSM in this research. They are:

- First is as a comparison to the intrinsic on-robot ranging sensor-based dynamic SSM implementation.
- Second is to use it as a ground truth to analyze the measurement accuracy of the on-robot ranging sensors.

The extrinsic sensors explored for the implementation of SSM are 3D depth camera-based

setup and a motion capture setup.

In the following sections, the results of the exploration to determine which form of extrinsic sensing (3D depth camera or motion capture setup) for dynamic SSM for this research are presented.

3.4 3D Depth Camera-based Setup

In this section, a 3D depth camera-based setup for monitoring the robot workplace is implemented, and the viability of using it for dynamic SSM as an extrinsic form of sensing is discussed. The depth camera used is a Microsoft Kinect v2, an RGB-D camera, where the depth is calculated using a structured Time-of-Flight camera. This setup is inspired by the work done by Morato et al. in [63] that uses four Kinect V1 sensors (older version of Microsoft Kinect) to perform human skeleton tracking in the 3D environment. As the Kinect V2 sensor provides a wider field of view and more robust depth measurement, the monitored workspace will be larger than the one presented in [63]. In this setup, three Kinect V2 sensors are used to monitor the workspace. The dimensions and the schematic of the monitoring system setup are shown in Figure 3.6. These dimensions are based on the recommendation of iPi Motion Capture software [24]. This area of monitoring is larger than the robot UR10's maximum workspace which is $2.6m$ in diameter.

The Kinect V2 provides depth information and also tracks human skeleton/body tracking points. The depth information can be used to generate a 3D pointcloud of the monitored workspace. In order to generate a 3D pointcloud, the steps required are camera calibration, depth data frame generation and synchronization, registration and merging of point-clouds in real-time. The generation of a merged and registered 3D pointcloud from multiple cameras is a complex and computationally intensive task [64]. The details are not discussed as it is not in the scope of this research, but the readers can refer to [62, 64] for more detail.

The experiment setup and the interface for the generation and synchronization of depth and body tracking information from multiple Kinect V2 is shown in the following Section.

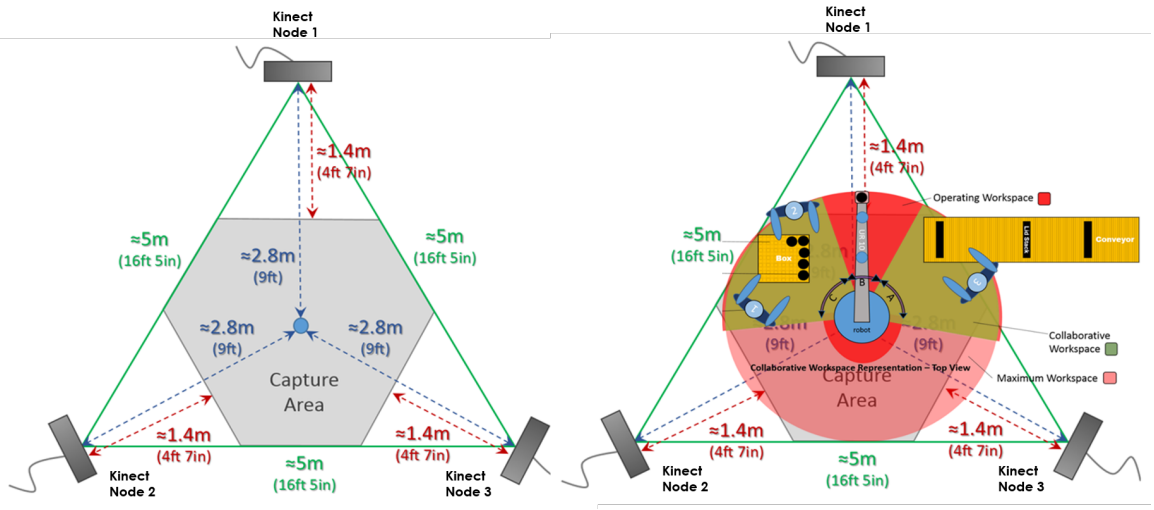


Figure 3.6: A schematic representing the experiment setup of the monitored collaborative workspace of the robot [7, 24].

3.4.1 Experiment Setup

In the experiment setup, two Kinect V2 cameras are used for testing the implementation of the setup. This testing setup implemented for the robot workspace of UR10 is shown in Figure 3.7. The experiment setup implemented pan-tilt mounts to adjusts the coverage of the Kinect sensors. The workspace covered by the sensors was also tested by simulating the sensor setup in V-REP (refer Figure 3.2). Due to software limitations imposed by the sensor, only a single Kinect V2 sensor can be interfaced with a computer with Windows OS. Hence, multiple computers with Kinect V2 sensors were connected over a local shared TCP/IP network. The data from each computer was shared using a server-client setup. An open-source JavaScript-based server environment called *Node.js* was used for transferring data over the network via web-sockets. In this setup, the master node (server) received all the camera information from *Kinect V2 Nodes* (clients). A Kinect V2 client node is defined as a sensor interfaced with a computer is transmitting over the network. The information available from the Kinect Nodes is listed in the following section.

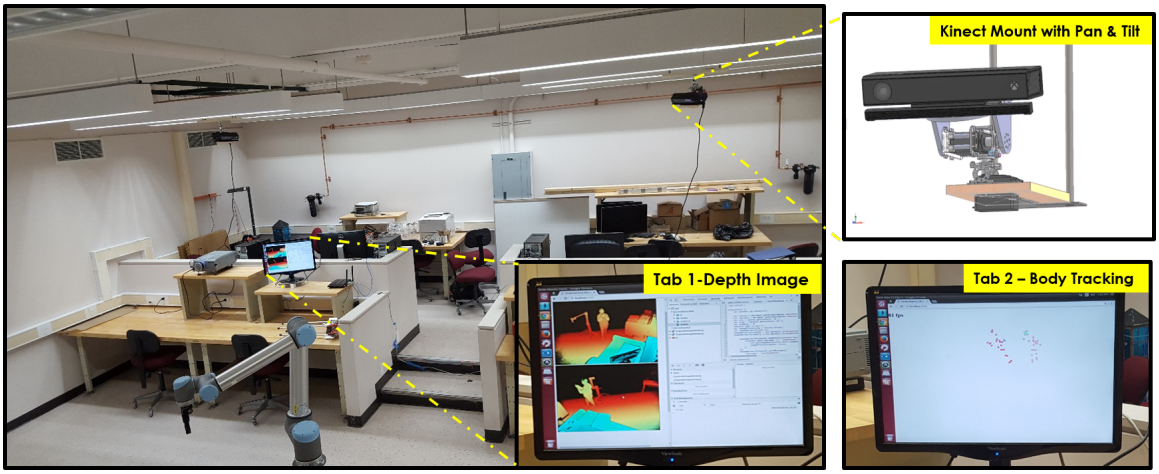


Figure 3.7: The Kinect testing setup used for monitoring the workspace. This image also shows the mounts to hang the Kinect v2 sensors from the ceiling. The output of the Kinect cameras showing the raw depth image data and body tracking information from two cameras are displayed in the two tabs of the browser.

3.4.2 Information from the 3D Depth Camera

The 3D depth camera, Kinect V2 is an RGB-D camera that provides a 1960×1280 RGB color image, a depth image and the position and orientation of human-body skeletal joints. Based on this information sources the Kinect Nodes provide a custom multi-sourced information that can be shared with the master node. The following information was available and shared from the from each Kinect v2 Node:

(a) **Body Tracking**: A JSON [1] message with position and orientation of 25 tracked body 3D points in Camera Space. It also provides information about the human body distance from floor, and distance from the camera.

(b) **Raw Depth / Pointcloud**: A 16-bit depth image of size 512×424 pixels at 30 frames per second (FPS), with each pixel value $0mm$ to $5000mm$ (can be more but accuracy drops) in camera space.

(c) **Color**: A high-definition RGB color image of size 1960×1280 pixels at 30 FPS.

(d) **Color+Body**: An RGB image with the tracked body points overlaid as a skeleton.

(e) **Pointcloud+Body**: A multi-source information of raw depth data and the tracked

[1] JavaScript Object Notation (JSON) is a syntax for storing and exchanging data over network. It is a text based, human-readable lightweight data-interchange message format.

body points in the depth frame.

(f) **Human Segmentation**: An RGB image with the tracked human body segmented.

(g) **Human Face Stabilization**: An RGB image of the tracked human body such that the image is scaled and cropped to ensure the human face is in the center of the image.

The results of these implementations are shown in Figure 3.8. This was done to understand the benefits and capabilities of using a Microsoft Kinect v2 as the 3D depth camera. Thus using an RGB-D camera provides contextually rich information that can be used in many ways for HRC. For the implementation of dynamic SSM, the information from the camera used is the **Pointcloud + Body** for the generation of a 3D depth map and tracking the human body for segmentation. The color image is used prior to the setup for camera calibration and registration of the Kinect v2 cameras. This information is used to represent the 3D points from all the Kinect v2 cameras in a common reference frame. The software architecture of the distributed system for combining the information from all the Kinect nodes is presented in the following section.

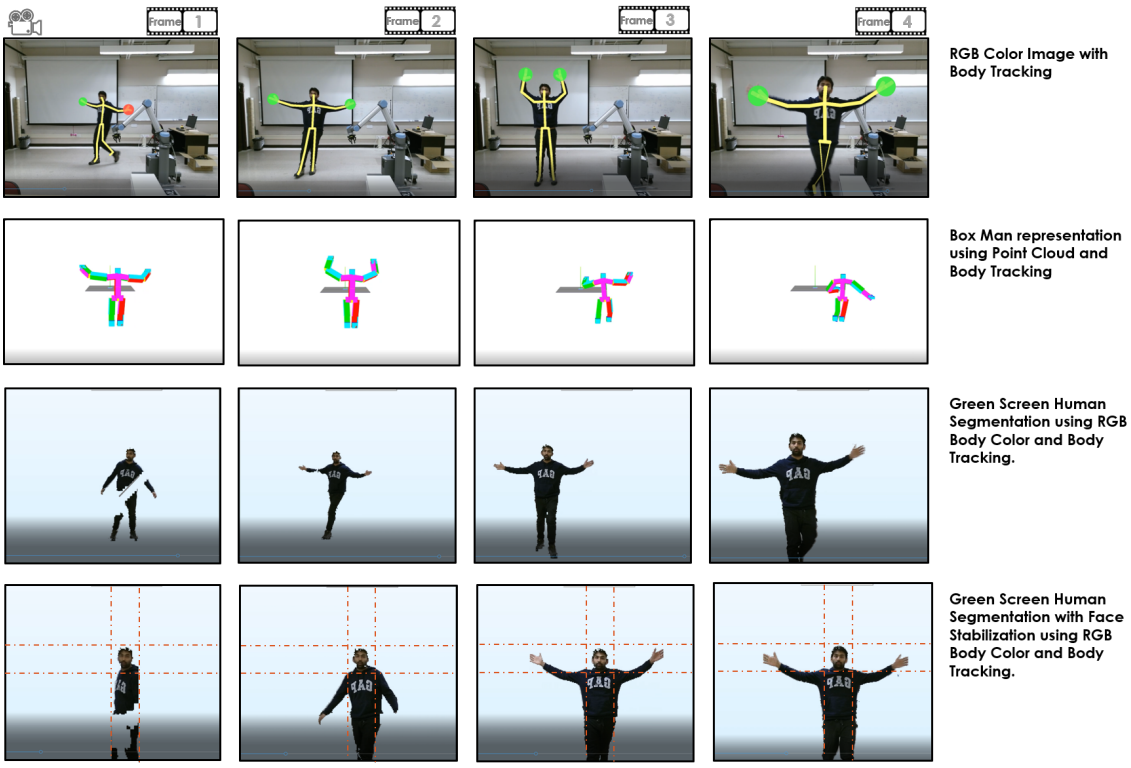


Figure 3.8: Information Available from Microsoft Kinect based setup.

3.4.3 Software Architecture for 3D Depth Camera-based System

In this section, the software architecture of the distributed system for a merged 3D point cloud registration is shown. This setup was also used to interface with the UR10 robot using ROS for the information needed for robot pointcloud segmentation [2]. The merged and registered point-clouds are shown in a web-browser using WebGL and Three.js. The first approach for the software implemented used web sockets to share information asynchronously with a master node. The data is then synchronized on the master node and using web workers, the 3D points registered, and merged to generate the point cloud in the browser. The block diagram of this architecture is shown in Figure 3.9(a). In this setup, the pointcloud data is merged with the *Node.js* server as the master node. This node communicates with the ROS server for robot information.

The second approach is based on using ROS as the middle-ware for the distributed system. The Node.js node is used as a relay to receive asynchronous data from the Kinect Nodes and publish it to ROS. This resulted in a more modular approach, and the synchronization of the frames was done using ROS.

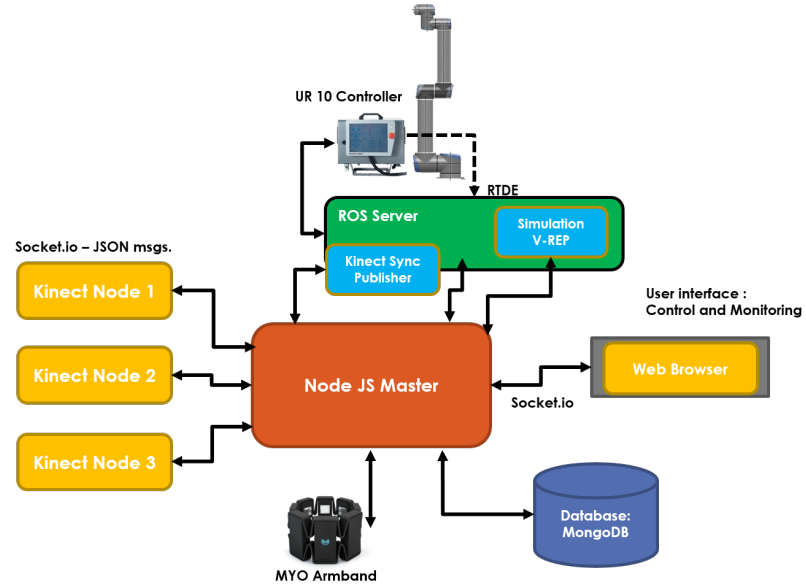
The results of these approaches for the setup are discussed in the following section.

3.4.4 Results for 3D Depth Camera-based System

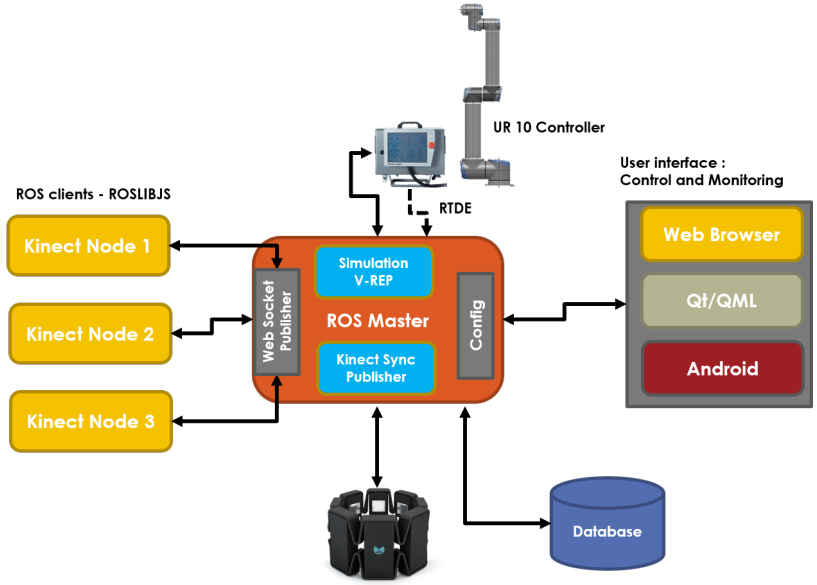
In this section, the resulting merged 3D point cloud registrations for the aforementioned approaches are shown in Figure 3.10. This shows frames of the real-time pointcloud registration and merging. The red pointcloud is from Kinect Node 1 and the green point cloud is from the data in Kinect Node 2. The blue and the magenta dots are the body tracking information from the Kinect nodes 1 and 2, respectively.

The results of the first architecture are shown in Figure 3.10(a). The point cloud from the cameras are aligned and registered; however, the body tracking information was not overlaid. This was due to the loss of synchronization for body tracking. Due to the asynchronous nature of the *Node.js* server environment, the Kinect based setup was slower and resulted in erroneous registration and merging of point-clouds. This resulted in a lag of

[2]The implementation of this setup is shared at https://spk4422.github.io/Multiple-Kinect-v2-Node.js-ROSlib.js—Project/



(a)



(b)

Figure 3.9: (a) System software architecture V1 setup for generating point clouds and merging from Microsoft Kinect Cameras (b) System software architecture setup for V2 for generating pointclouds and merging from two Microsoft Kinect Cameras.

frame updates. The frame rate reported was less than 10 FPS. One of the reasons was the amount of data being shared on the network. The local network had 100 Mbps (megabits per second) network bandwidth, which resulted in a bottleneck for the transfer of data. It can be observed in video Frame 2 and 3 of Figure 3.10(a) that the point-clouds are not aligned and the synchronization fails due to the lag or the availability of data from one of the Kinects.

The results of the second architecture are shown in Figure 3.10(b). Using ROS [107] as the middleware for this system, the load of computation was distributed. There was relatively less lag in comparison to the first approach. As shown in Figure 3.10(b) frame 3, in case of occlusion, the human body trackers are not accurately aligned with the 3D points of the human. Moreover, as ROS being the master server, the UR10 robot data was also represented in the same space. The 3D pointcloud of the workspace was generated at greater than 15 FPS. The network was also updated to 1000 Mbps, which resulted in the pointcloud update rate at the max of 30 FPS for two Kinects. Thereby resulting in a real-time 3D merged pointcloud generation and monitoring of the workspace.

In the next section, we discuss the pros and cons of using a 3D depth camera-based system as an extrinsic form of sensing in HRC applications.

3.4.5 Pros and Cons

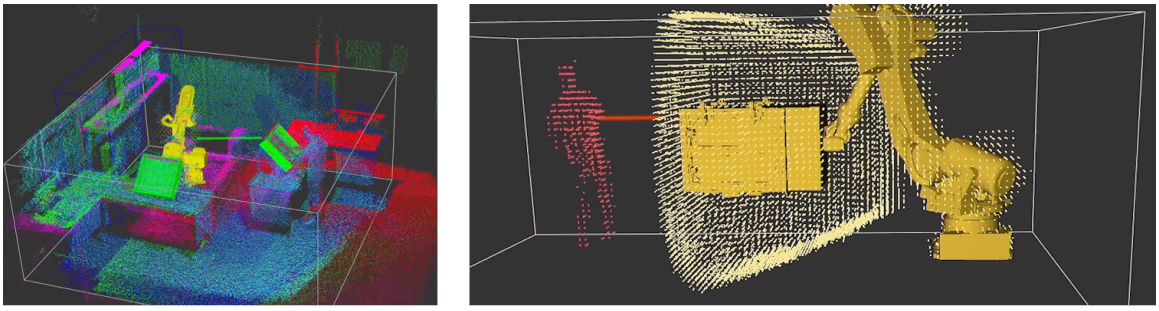
The 3D depth camera, Kinect V2 provides depth information, color image, and also tracks human skeleton points. The depth information is used to generate a 3D pointcloud of the monitored workspace. The information provided by using multiple Kinect v2 cameras is contextually rich and can be used in HRC. The generation of a merged and registered 3D pointcloud from multiple depth cameras is a complex and computationally intensive task [44, 62]. Following the generation of 3D point-clouds, the 3D points occupied by the human and robot are to be segmented. The occupancy of the human in this 3D pointcloud is determined using the body tracked points provided by all the Kinect v2. The tracking accuracy decreases when the human is occluded. The robot's known 3D CAD model and its kinematic pose, as reported by the robot, can be used for point-cloud segmentation of the robot. This is also a non-trivial problem and adds to the complexity of the setup. The information required from this 3D depth camera setup is the minimum human-robot

Figure 3.10: (a) The resulting point cloud registration. This shows frames of the real-time pointcloud registration and merging using the System V1. Video Link https://youtu.be/kO0drvHSpGA (b) The resulting point cloud registration. This shows frames of the real-time pointcloud registration and merging using the System V2. Video Link https://youtu.be/-Sv2Gansipw

distance in 3D space for the implementation of dynamic SSM. The segmented 3D points can then be used for measuring the minimum separation distance. This added complexity would affect the real-time performance of the system, which would negatively impact the overall safety of the HRC setup.

A novel depth-space approach for the calculation of minimum separation distance is presented in [60]. The computational load for a single camera is relatively less, but as the number of cameras increase, so does the load and also the data being transmitted over the network. In order to scale this setup, the use of parallel computing and GPUs would be required.

A similar approach for using 3D point-clouds for dynamic SSM has been implemented by a company called VeoRobotics [25]. The merged 3D point clouds from ToF based cameras are shown in Figure 3.11(a). The different colors represent the depth information from different ToF cameras registered and merged. An occupancy grid map is generated following the segmentation of human and the robot 3D points. The robot 3D CAD model is overlaid and updated using the joint poses reported by the robot controller. The red line indicates the minimum separation distance measured between the point clouds.



(a) (b)

Figure 3.11: (a) Merged Point Clouds from ToF based Camera. Different colors represent the depth information from different ToF Cameras. (b) Occupancy Grid map generated following the segmentation of human and the robot 3D points. The robot cad model is overlaid and updated using the joint poses reported by the robot controller. The red line indicates the minimum separation distance measured. – (courtesy of VeoRobotics Inc. [25]). Video Link: https://youtu.be/NXGqMCA5gn8

Thus, the 3D depth camera-based approach provides a detailed picture of the workspace with a high computation cost. The implementation of this setup itself is a large body of work, and it would digress from the objective of this research of using intrinsic sensors for

SSM. Hence, this approach was not chosen as the form for extrinsic sensing. In comparison, using a third party motion capture software that tracks markers on the human would result in similar information. It would be relatively less computationally expensive as the number of points tracked is limited. The following section describes the setup used for human tracking and robot pose representation using a motion capture system.

3.5 Motion Capture Setup

This setup uses a camera to track optical markers placed on the human, robot or objects of interest to track their position in 3D space. Instead of a 3D point cloud, the markers are localized from a given reference-frame. This form of 3D localization is highly accurate and can be used as ground-truth. A Nature-point OptiTrack motion-capture system is used in this research [68]. A set of eight Flex13 motion tracking cameras are used to monitor a workspace of size $7.3m$(24ft) $\times$ $6m$(20ft) (also refer Figure 2.13(b)). A software provided by the Optitrack called Motive provides tracking of the markers at 120FPS [68]. The information provided by the Motion Capture system is not contextually rich in comparison to the 3D depth-based setup, but it can provide highly accurate 3D localization of retro-reflective markers on the entities to track.

The mocap system can track the position and orientation of a rigid body as a group of multiple markers in a 3D space. It uses the formation of these grouped markers to represent a rigid body in 3D space. In this research, these rigid-body markers are placed on a human body to track its motion and pose in 3D space. However, the localization of a human alone is not enough representation for the information necessary for calculation of the human-robot minimum distance. This information needs to be represented with the robot kinematics and pose in the same reference frame to calculate the minimum separation distance. This was achieved by representing the robot base as a rigid body for its localization with respect to the world frame. The restricted/static workspace such as tables and other static objects in the workspace were mapped in the 3D workspace using these markers.

For tracking the robot motion in 3D space in the workspace, there were two approaches. The first is to use markers on the robot links and represent the robot links as rigid bodies moving in space. Second is to use the 3D CAD model of the UR10 robot available for a robot in simulation, and use the internal joint data i.e. proprioceptive information of

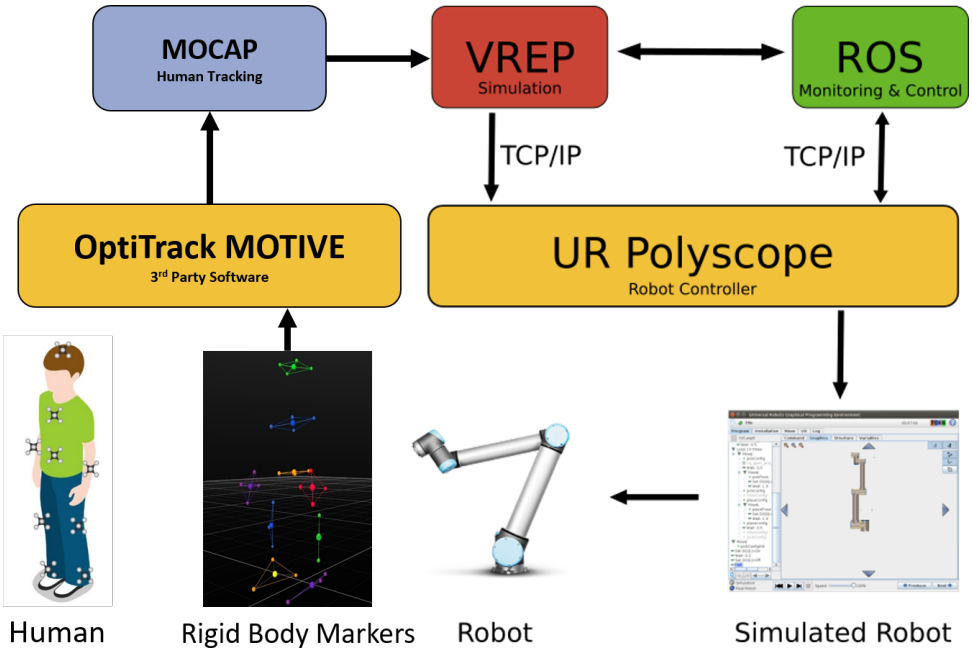


Figure 3.12: A block diagram showing the motion capture setup used with the digital-twin setup for combining the human and robot localization with respect to a common frame.

joint poses to update the CAD model in a *simulation-in-the-loop*, which is the digital-twin setup. Both are valid approaches and have been used for tracking in robotics research [1, 66, 113, 114]. In fact many research use markers for quad-copters and mobile robots for tracking them in 3D space.

However, as in this research, a digital-twin setup was used, and the robot pose information was available at 125 Hz from the robot controller and is more accurate information of the robot; the second approach was chosen. Thus a digital-twin setup for tracking human determine the static workspace and objects was preferred. The block diagram of the motion-capture and the digital-twin setup is shown in Figure 3.12.

In this research, V-REP is used to model the workspace for the HRC setup. The human tracking and the robot pose information is used to update the digital-twin, i.e. the simulation in real-time. The V-REP simulation is graphically updated at 20 FPS. The human tracking information is used to update a human-avatar pose and location in the virtual world. Having represented the human, robot, and the workspace in real-time, the inter-mesh

[3] distances can be calculated between all entities virtually represented in the digital-twin setup. This information can then be used for implementation of dynamic SSM. Although there is computation involved, this system is accurately able to represent and mimic the physical/real state of the workspace due to the tracking accuracy of the mocap system and integration of the robot pose data. This synthetic representation can be considered an *ideal* virtual representation of the physical world setting.

Thus, Motion Capture was the preferred choice for the extrinsic perception sensor for implementing dynamic SSM. Also, this sensor setup provided the ground-truth for measuring human-robot distance because of its tracking accuracy of rigid bodies in the workspace.

3.5.1 Results for the Motion Capture Setup

The implemented setup used in this research is shown in Figure 3.13. The human has rigid body markers placed on the chest, arms, and legs. A humanoid avatar is controlled based on the human body kinematics for its pose, and the location of the markers is used to localize the avatar. The simulated robot pose is updated based on the movement of the UR10 robot in the physical world. The mesh to mesh distances between the closest point of the robot and the humanoid avatar is taken as an estimate of the minimum separation distance between the human and the robot.

Next, we present the pros and cons of using the motion-capture system as an extrinsic form of sensing in HRC applications.

3.5.2 Pros and Cons

The motion capture setup, unlike the 3D points tracks only one marker in the 3D space. The localization of this setup is computationally less expensive in compassion to the 3D depth setup. Unlike the 3D depth cameras, this setup does not provide contextual information about the workspace. It is unable to represent other aspects of the environment, such as the restricted workspace of the robot, the overall shape of a human, or the kinematics and dimensions of the robot. It needs to be used in conjunction with a digital-twin setup. The workspace covered by this setup is larger than the 3D camera setup. The tracking using the

[3] A mesh is a collection of vertices, edges, and faces that describe the shape of a 3D object

Figure 3.13: Mocap result of the digital-twin implemented. It tracks the human using rigid body markers and the robot pose is updated in the simulation using the information reported by the UR10 Robot. Video Link

markers is more accurate using the mocap system than 3D cameras. The mocap system can track entities at a higher rate in comparison to the 3D depth camera setup.

The motion capture system setup uses reflective markers on the entities for tracking in the workspace. In an industrial setting, using rigid body markers on human can be cumbersome. Rigid body markers are not generally used for tracking human pose. A body suit with markers is used to accurately represent human joint position and orientation as shown in Figure 3.14. The body tracking would give information regarding the human kinematics which can then be used to represent the human body pose very accurately in HRC setups. This will be essential if human actions are to be learned and predicted. For spatial and temporal information, rigid markers are enough for the implementation of dynamic SSM. As this is an extrinsic sensor setup, it suffers from loss of tracking due to occlusion of markers. This would result in a loss of tracking. Hence cameras need to be positioned for maximum coverage. For every change in the placement of cameras result in a time-consuming calibration procedure to ensure complete coverage of the workspace.

Next, we present details of the Time-of-Flight sensor used as the on-robot ranging sensor for implementation of dynamic SSM.

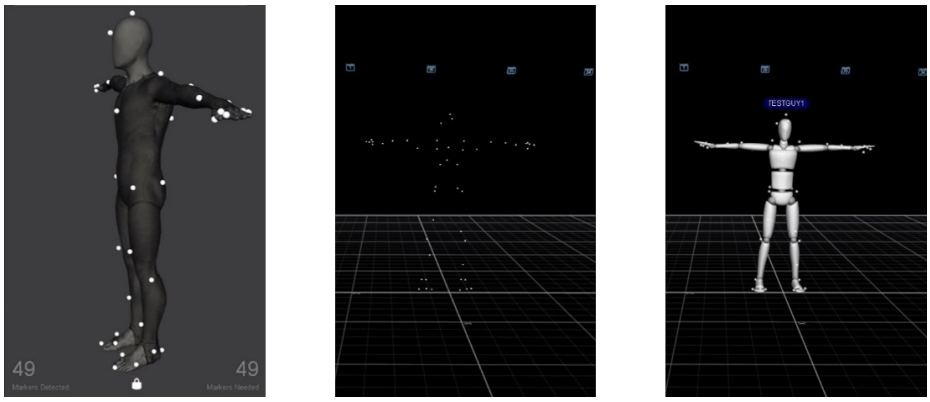


Figure 3.14: Human body tracking using bodysuit with markers. The software overlays a human avatar.

3.6 Time-of-Flight Laser Ranging Sensor Array

A Time-of-Flight (ToF) laser-ranging sensor is chosen for implementing intrinsic or on-robot based dynamic SSM. The basic element of the ToF sensor is based on the VL53L0X/VL53L1X architecture by ST Microelectronics [26, 27]. SPADs (Single Photon Avalanche Diodes) or photon-counting Avalanche Photo-Diodes (APDs) are at the core of the VL53L0X/VL53L1X architecture [115]. This is a single unit, direct time-of-flight laser ranging sensor module with high accuracy and narrow field of view. The distance to an object is determined by time measurement between emitted and received light pulse (infrared laser source). It is referred to as the direct Time-of-Flight (ToF) measurement of optical pulses sent by a light source [116]. The distance is retrieved based on the known speed of light. For instance, an object at $1cm$ from the system would result in a $66ps$ (picoseconds) time shift between the emitter and the receiver. This approach enables rapid, accurate distance measurements independent of the characteristics of the target object. Furthermore, the $940nm$ emitter operating in the non-visible spectrum eliminates distracting light emission. Thus, being immune to ambient illumination and optical path variations, the sensor consistently measures the same distance.

The VL53L0X architecture based ToF sensor is a single SPAD based measurement unit [26]. It can detect up to $2m$ with $0.001m$ resolution and a FOV angle of 25° degrees. The optimal/effective operating range for detecting distance robustly is between $0.03m$ to $1.2m$. The sensor's accuracy is specified to range from $\pm 3\%$ at best to over $\pm 10\%$ in

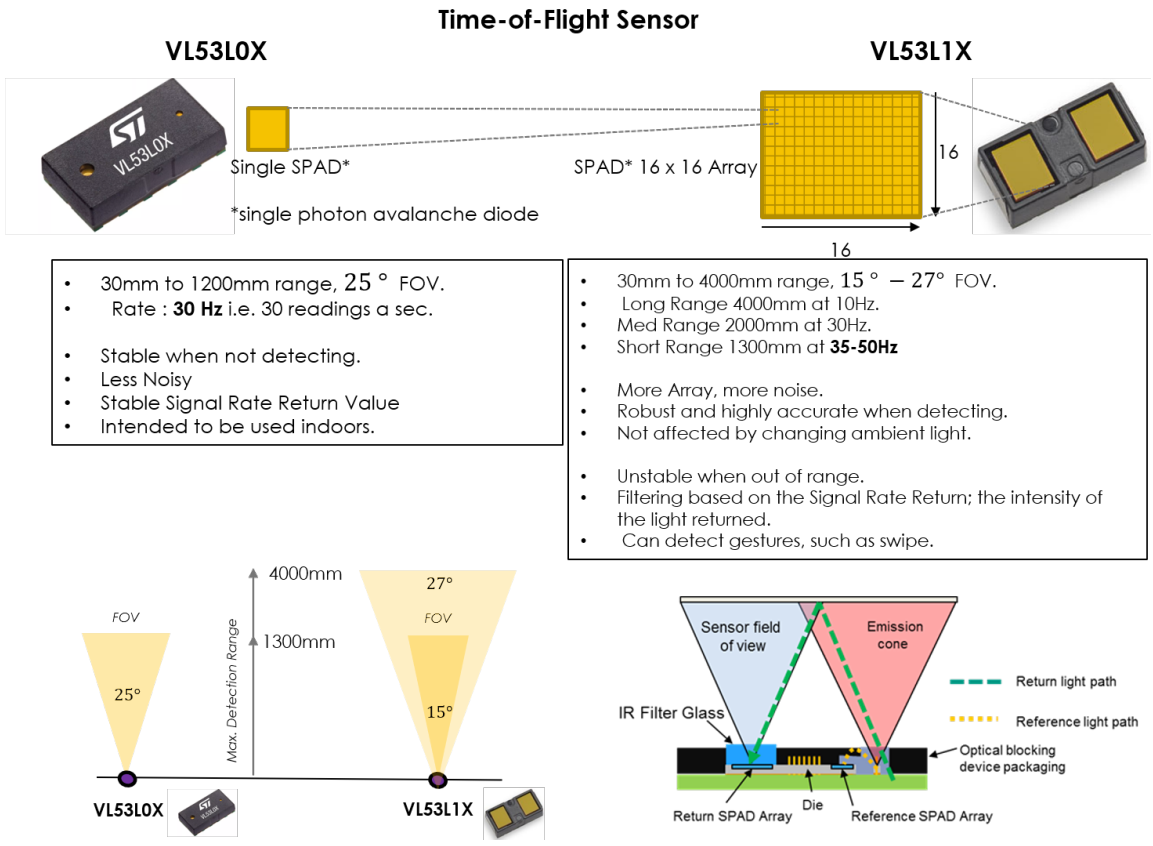


Figure 3.15: The specification and characteristics of the Time-of-Flight sensor modules VL53L0X [26] and VL53L1X [27] used as the basic sensing element of a ToF sensor node.

less optimal conditions [26, 116]. This sensor module was used in the initial phases of testing the ToF based sensor array. In the later stages of the ToF sensor array prototype development, this was updated to the VL53L1X architecture based ToF sensor [27]. The VL53L1X architecture contained a 16×16 SPAD array with a longer measurement range ($4m$) and a higher sampling rate ($35Hz - 50Hz$). The differences and features of the two ToF architectures are shown in Figure 3.15.

The VL53L0X/VL53L1X ToF sensor is a smart sensor that can be interfaced using I2C communication protocol. It has a programmable I2C slave address and multiple sensors can be daisy-chained for communication of distance information. In addition to the distance ToF distance measurements, the sensor also provides accuracy (confidence) of measurement, signal strength, and error codes [26]. More information about the data available from the ToF sensor is presented in Chapter 6.

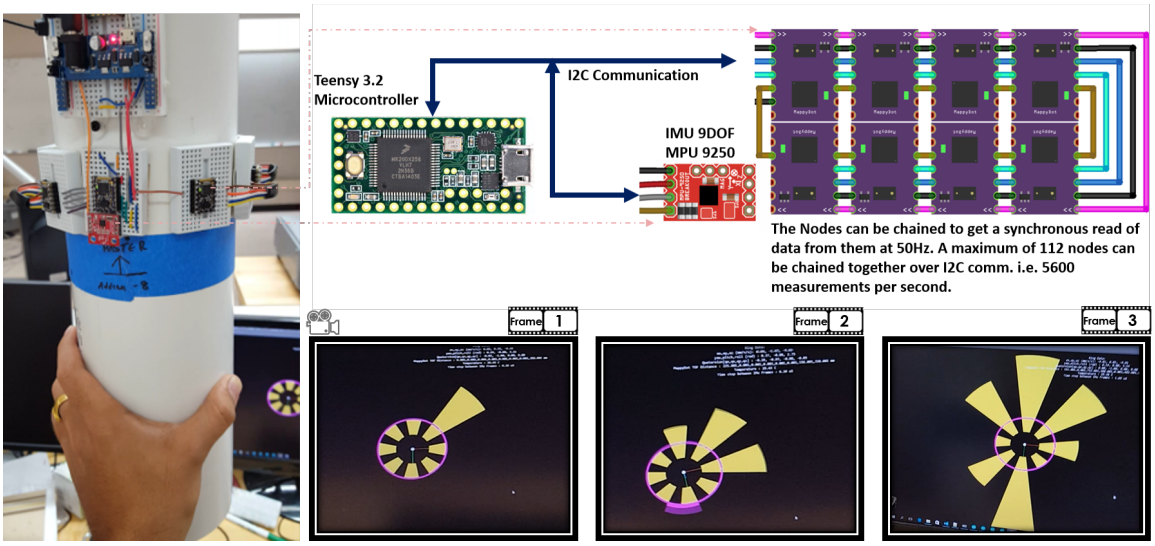


Figure 3.16: The implementation of sensor arrays chained together to represent distance around a cylindrical object. The result is shown in the video frames https://youtu.be/JeL_Dpxn5tA.

A ToF sensor node consists of an off-the-shelf MappyDot board that uses an Atmega328PB Micro-controller and a VL53L0X ToF module on each node to perform synchronous data collection, filtering and other operations for cleaner and robust distance data [117]. The MappyDot ToF sensor board has a very small footprint of $17.8mm \times 13mm$ (refer Figure 2.14(a)), smaller than a quarter dollar coin, which has a diameter of $24.26mm$. For testing the ToF sensor array, eight ToF sensor nodes are daisy-chained on an I2C bus and mounted on a cylindrical object equally spaced (radially) 45° degrees apart as shown in Figure 3.16. The ToF sensor array is connected to Teensy 3.2 micro-controller and also a 9DOF IMU Invensense MPU9250 for acceleration and gyroscope readings. The IMU data can be used to determine orientation and direction of motion of the cylindrical object or the robot link it is placed on in 3D space [118, 119].

The implementation of sensor arrays chained together to represent distance around a cylindrical object using the existing Node.js server environment. The output of the IMU and ranging response from the sensors was tested by plotting a 3D disc object and expanding sectors based on the distance measured by the sensors in real-time. The result is shown in the video frames shown in Figure 3.16. The distance measurement was updated every $45ms$.

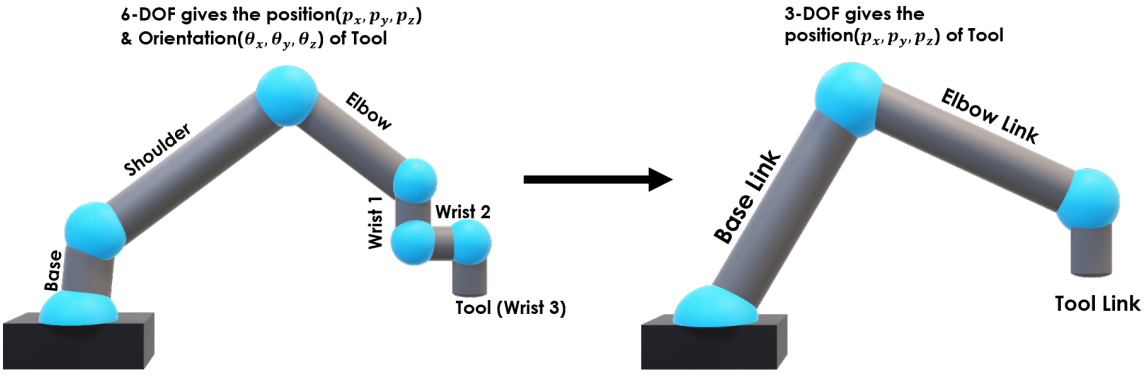


Figure 3.17: An Geometrical Approximation of Ellipsoids for a 6 DOF robot such as the UR10. Ellipsoids (Base, Elbow and Tool-end effector) for 3 major links of the robot.

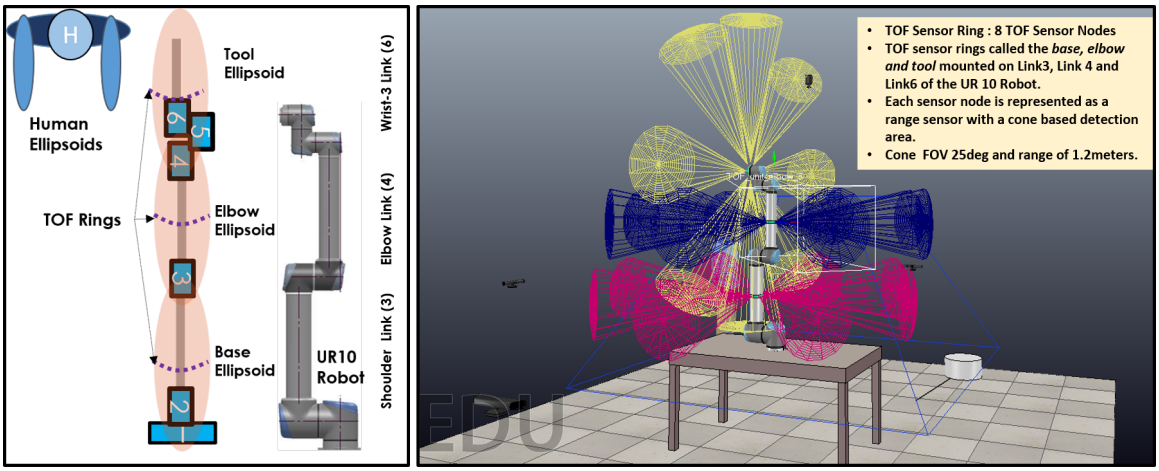
In the next section, we discus the placement of the ToF sensor array on the robot for intrinsic sensing.

3.6.1 Placement of ToF Sensor Array

Unlike the work done in [20, 58], where infrared (IR) triangulation sensors are strategically placed on the robot links based on the optimization of the kinetostatic danger field, we propose a simpler strategy that would be easier to implement with robot manipulators in industry. The commonly used safety strategies are potential fields [60, 120], kinetostatic fields [71] and geometrical approximations such as spheres and ellipsoids [64, 82, 84]. For potential field and kinetostatic field-based approaches, the minimum distance is calculated between the key points on the robot and the nearest point on the human. But extra computational power is required for searching the closest point on the human. Hence, geometrical approximations have been used to prune the search space by making approximations of robot links and human limbs/body parts and calculating the human-robot minimum separation distance.

In this research, based on the methodology used in [82], a six degree of freedom (DOF) robot such as the UR10 is approximated as a three degrees of freedom robot as shown in Figure 3.17. A 6-DOF robot represents the position ($\left[p_x, p_y, p_z\right]^T$) and orientation ($\left[\theta_x, \theta_y, \theta_z\right]^T$) of the end-effector/tool using its forward kinematics for joint angles ($\boldsymbol{q}$). The approximate 3-DOF robot represents only the position of the end-effector ($\left[p_x, p_y, p_z\right]^T$).

This is done by using ellipsoidal approximation as shown in Figure 3.18 [82] . This



(a) (b)

Figure 3.18: (a) An Geometrical Approximation of Ellipsoids for a 6 DOF robot, here the UR10. Ellipsoids (Base, Elbow and Tool-end effector) for 3 major links of the robot, and its centers where the ToF sensor array rings could be placed.(b) A simulation depicting the detection volume and coverage of sensors mounted on the robot https://youtu.be/s2tmIluRh5g.

is discussed further in the following Chapter 4. A 6-DOF freedom robot is represented as using three ellipsoids that constitute the reach of the robot. The ToF sensor arrays are placed at approximate centers of the ellipsoid approximations, referred to as *base*, *elbow*, and *tool*. The minimum distance is calculated from the centers of these ellipsoids, which coincide with the centers of the *shoulder*, *elbow*, and *wrist-3* links of the UR10 robot. Hence, the ToF sensor arrays are placed similar to a 'watchband' or as a ring at the center of these links. This form of placement is easier and simpler to mount.

In the digital-twin simulation using V-REP, an emulated cone-shaped proximity sensor model with a FOV of 25° degrees and 1.3m detection range is used to simulate the behavior of the VL53L0X / VL53L1X ToF sensor node [29, 76]. These are placed at the centers of the UR10 robot links to visualize the sensing coverage, as shown in Figure 3.18.

A test setup to compare the distance measurement accuracy of a single VL53L0X ToF sensor, the simulated ToF proximity sensor, the mesh-mesh distances in the digital-twin simulation, and the distance using rigid-body markers with motion-capture setup is presented in the next section.

3.6.2 Comparing Motion-capture, Digital-Twin and Time-of-Flight Distance Measurements

The test setup is designed to measure the distance between a sensing board with the VL53L0X ToF sensor node and a flat white-colored object. The test setup is shown in Figure 3.19. A rigid-body mocap marker is placed on the object and the sensor board, which is then localized using the motion-capture setup. A measuring ruler is placed between the sensor board and the object to get an accurate distance. The ruler is used to calibrate the distances by taking into account the offsets due to the thickness of the object and the sensor board. The sensor modalities compared in the test setup for the distance measurement are detailed as follows:

- **Motion-Capture (Mocap)** based distance is calculated by taking the euclidean (L2norm) distance between the positions of the rigid body markers placed on the object and the sensor board. The mocap system updates the positions of the rigid-body markers every $8.33ms$ or $120Hz$. There is minimal computation required for calculating the distance between two points of the rigid body markers.
- **VL53L0X ToF sensor node (ToF)** reports the measured distance by the ToF sensor connected to a Teensy 3.2. micro-controller over I2C on the sensor board that relays the distance measurement over serial communication. The sampling period is set to $45ms$.
- **Mesh-Mesh Distance in Digital-Twin (Ideal)** is between the simulated 3D shaped of the sensor node and the object that is localized in the digital-twin simulation using the motion capture.
- **Simulated ToF sensor in Digital-Twin (Sim)** is the distance measured using the simulated cone-shaped proximity sensor positioned on the sensor board and the object.

During a test, the object board is moved on a linear slide between $0.6m$ to $1m$, and the distance measurements from all sensor modalities recorded.

The mesh-mesh or the *Ideal* distance between the simulated object and the sensor board is the euclidean distance between the closest points on the mesh of their corresponding 3D shapes in the digital-twin simulation. Thus, the smallest change and error in the orientation of the simulated object and sensor board could result in a change to the minimum distance. On the other hand, the emulated cone-shaped proximity sensor performs ray-cast (measurement from a point light source) based measurements from a single point on the

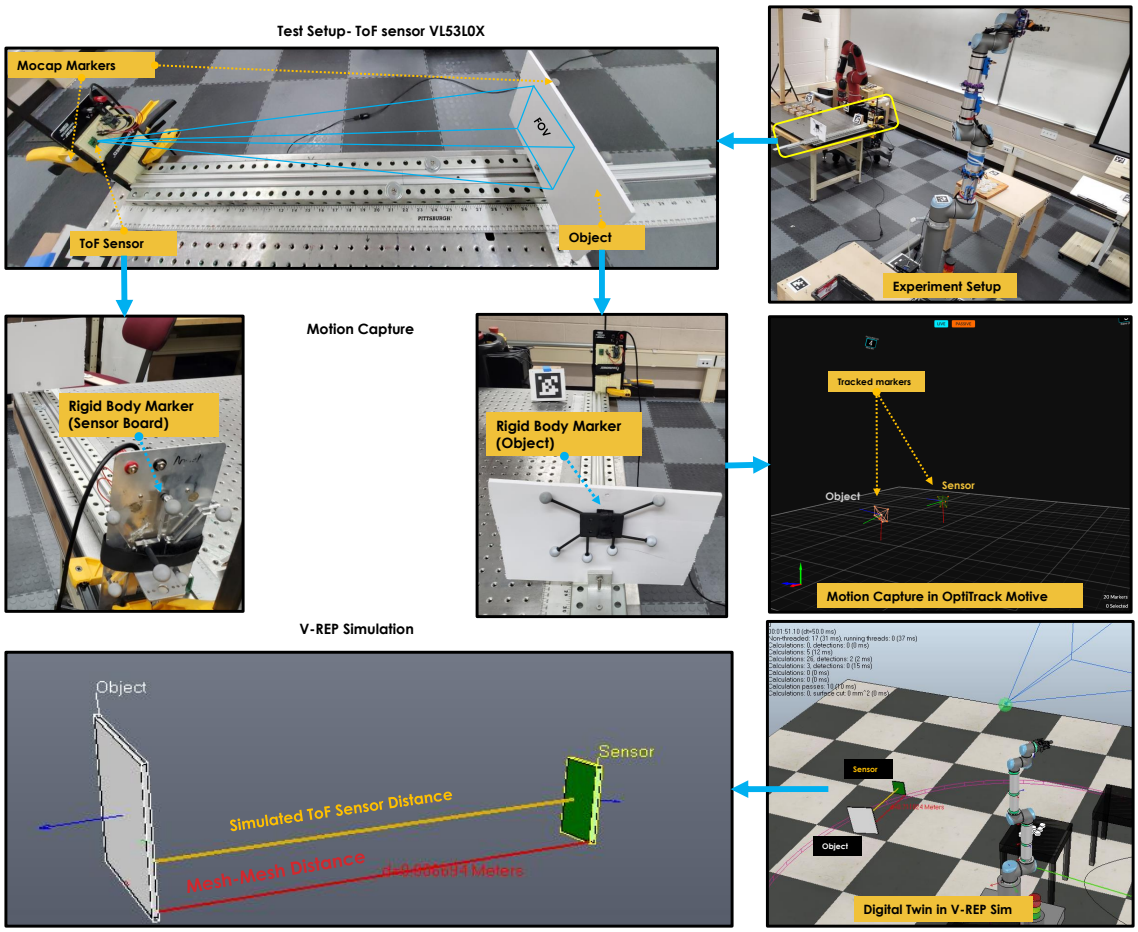


Figure 3.19: A diagram showing an experiment setup for comparing the accuracy and update rate for distance measurements using mocap markers, a digital twin, and a ToF sensor. Video Link

sensor board and all the points that are in its field-of-view. Thus, as the number of mesh-mesh distance calculations in the digital-twin increase, the overall computational load also increase. In order to test the effect of increasing the mesh-mesh calculations, two digital-twin test setups with 'no-load' and 'load,' as shown in Figure 3.20, were used. In the 'no-load' setup, one simulated ToF sensor and one mesh-mesh distance calculations are performed. In the 'with-load' setup, a UR10 robot with three 8-node simulated sensor arrays (which is 24 proximity sensors in total), a simulated LiDAR proximity sensor, and two mesh-mesh distance calculations were performed in addition to the 'no-load' setup. This resulted in a total of 26 proximity sensor and three mesh-mesh distance calculations. This setup simulated the digital-twin of the robot workspace used in this research for dynamic

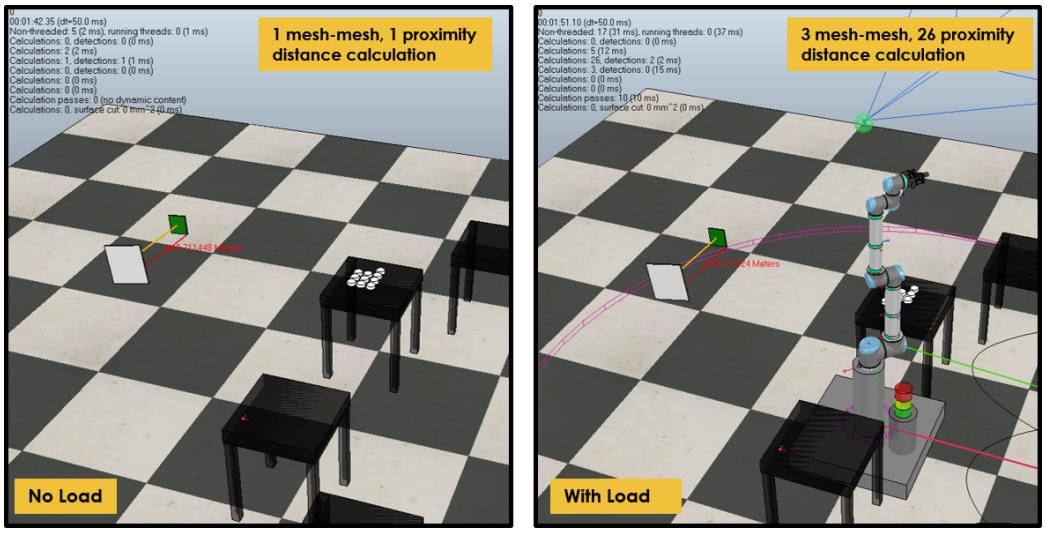


Figure 3.20: The digital-twin setup for no load condition where only 1 proximity sensor and 1 mesh-mesh calculation is performed (Left); and with load where 26 proximity and 3 mesh-mesh distance calculations are performed (Right) in the simulation.

SSM implementation.

The accuracy of distance measurements for all sensor modalities is very good. All the distance reported were within $\pm 30mm$ of the actual distance. The motion-capture setup reported the distance very consistently with a minimum standard deviation. The test setup with higher computation load resulted in quantization error and lag in the proximity sensor distance and the mesh-mesh distance measurements in comparison to the motion capture distance, as shown in Figure 3.21. The results of the update rates of the minimum distance measurements are shown in Table 3.1. The mesh-mesh (Ideal) distance is smaller because it determines the closest point on the object and takes into account the object orientation. A small misalignment in the placement of the object resulted in it not being parallel to the sensor board. The resulting detection was not from the center but a different closest point on the mesh. This form of distance measurement can be considered absolute or *Ideal* and thus was chosen for representing the ground truth minimum distance.

On the other hand, the distance measurement from the ToF sensor and the motion capture is unaffected. The complexity of calculation of the minimum distance is linear ($O(n_{ToF})$) and is proportional to the number of sensors (n_{ToF})), as the sensor directly calculates the minimum distance. Whereas, the mesh-mesh distance [29, 121] complexity ($O(n_{r,h}\log(n_{r,h}))$) is proportional to the number of points in a mesh representing the

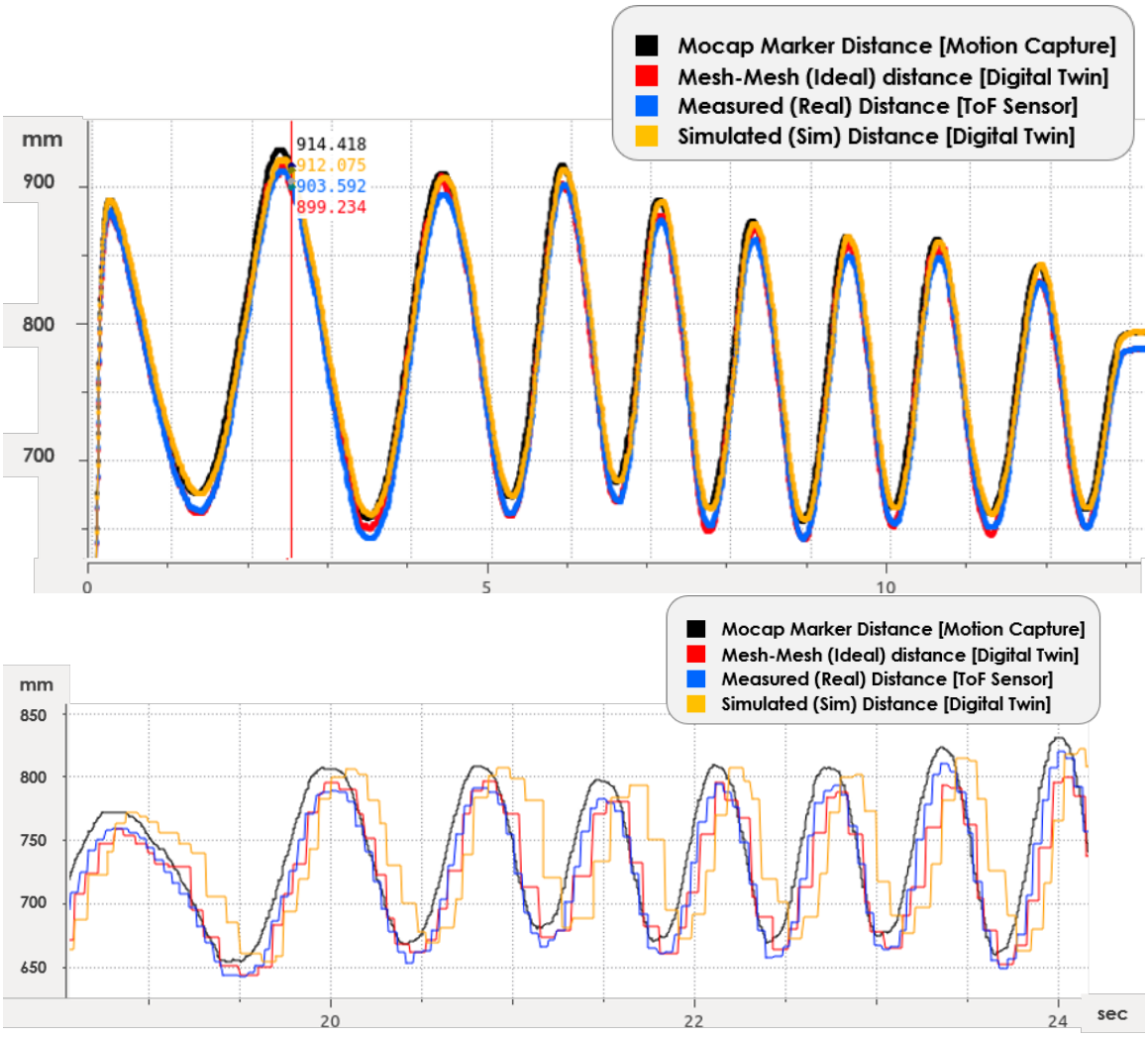


Figure 3.21: (Top) The comparison of accuracy of the distance measurements for motion capture, digital-twin, and ToF sensor. (Bottom) the quantization error and the lag in the update rate of measurements from the digital-twin as the computational load increases.

Table 3.1: A table comparing the update rate for the ranging measurements using rigid body markers digital-twin and a ToF sensor, with and without load in Digital-Twin

Sensing Modality	**Update Rate for 1000 samples in ms (Avg. (std.))**	
	No Load in Digital-Twin	**With Load in Digital-Twin**
Rigid Body Markers (Mocap)	8.33 (± 2.25)	8.327 (± 2.21)
Ranging Sensor (ToF)	45.98 (± 0.41)	45.99 (± 0.57)
Mesh-Mesh (Digital-Twin)	16.66 (± 0.62)	71.09 (± 18.84)
Simulated ToF sensor (Digital-Twin)	16.73 (± 2.08)	71.12 (± 18.42)

robot and human ($n_{r,h} >> n_{ToF}$) as it performs the search for the closest points between the meshes [122]. To summarize, the mesh-mesh minimum distance measurement in the digital-twin represents the advantages and the drawbacks of extrinsic sensing. The accuracy of the minimum distance measurement is higher and the update rate suffers by increasing the computational load. Hence, the digital-twin using the motion-capture setup is used to compare extrinsic and intrinsic sensing-based implementation of dynamic SSM safety measure.

In the next section, we present the pros and cons of using ToF sensor array.

3.6.3 Pros and Cons

The ToF sensor array provides no contextual information. But they provide a less computationally expensive option to measure distance from the robot to the human or object in the workspace. The accuracy of the ToF sensor node is accurate and within $\pm 30mm$ of the actual distance with a consistent update rate. As the ToF sensor is light-based, it was accurately simulated in V-REP using a cone-shaped proximity sensor. The ToF sensor, however, is less accurate in comparison to the motion-capture setup using rigid body markers. Moreover, the distance reading of a point on the object in the field-of-view is averaged. The eight-node ToF sensor array would have blind spots and lost coverage that would affect the minimum distance measurement between the human and the center of the robot links. An analysis of the minimum distance accuracy using the ToF sensor arrays has been presented in the following Chapter 4.

Next, we summarize the details of the exploration of components and sensors needed for the implementation of a dynamic safety measure in HRC.

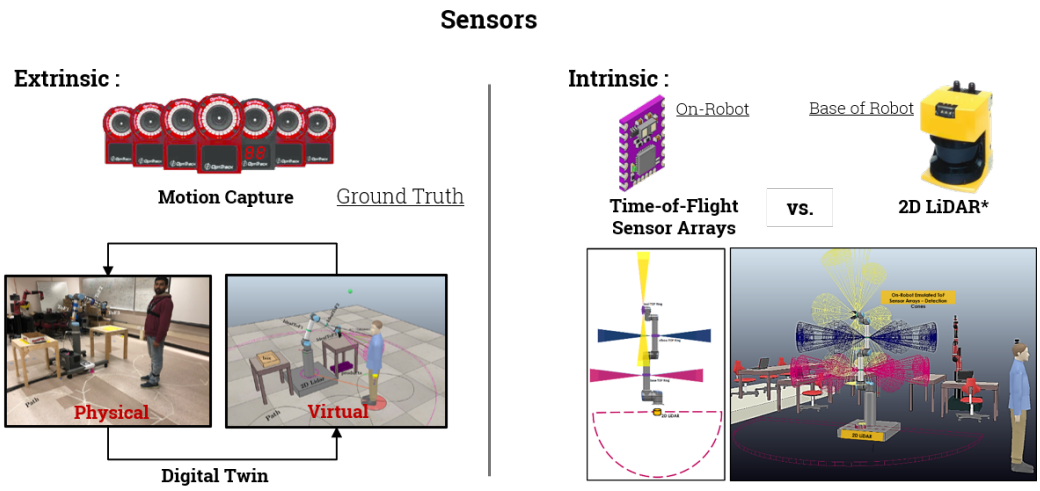


Figure 3.22: A figure summarizing the sensors chosen for the implementation of dynamic Speed and Separation Monitoring in this research.

3.7 Summary

A framework for a robot supervision system that highlights the components and subsystems needed to achieve dynamic safety during HRC is presented. This framework contains six subsystems: PERCEPTION, KNOWLEDGE, VIRTUAL WORLD, CONTROL, INTELLIGENCE, and INTERFACE. A *digital-twin* setup using V-REP is used as the VIRTUAL WORLD subsystem. An SSM based pre-collision control is chosen as the form of CONTROL. The INTELLIGENCE subsystem is used to implement the dynamic nature of the safety configuration for intrinsic based sensing. The ROBOT INTERFACE for a UR10 robot is done using existing a custom 'Real Time Data Exchange' protocol library and ROS support for UR10 robots for real-time monitoring and control of robot speed for SSM safety configuration.

For PERCEPTION, extrinsic sensors such as the Microsoft Kinect v2 camera and OptiTrack Motion Capture system are compared for human tracking and calculation of human-robot separation distance and velocities. The OptiTrack mocap system [24] is selected as the form of extrinsic sensing due its lower computational load, accuracy, faster implementation, and integration with the digital-twin setup. The conventional 2D scanning LiDAR placed at the base of the robot and a Time-of-Flight sensor arrays placed on the robot center links are chosen as intrinsic sensors for the implementation of SSM safety configuration (see Figure 3.22).

Table 3.2: A comparison of the 3D depth camera, motion capture and ToF based setups used for the implementation of dynamic SSM.

3D Depth Camera	**Motion Capture**	**ToF based Setup**
Contextually rich information, more detail of the workspace.	No contextual information.	No contextual information.
No explicit markers needed on, humans, robots, or objects in the workspace for tracking.	Markers needed for the tracked bodies in the workspace.	No markers are needed for the tracked bodies.
Computationally very expensive	Computationally less expensive compared to 3D cameras.	Computationally least expensive.
Less accurate in comparison to Motion capture system.	Highly accurate in comparison to the 3D depth camera system	Less accurate in comparison to Mocap and 3D depth cameras.
Less coverage in comparison to Mocap	Largest Workspace Coverage.	Coverage around the robot.
Calibration required	Calibration required	No calibration required.
Less Costly ($$) in	Very Costly ($$$)	Low Cost ($)

A comparison of the setups explored for the implementation of dynamic SSM is shown in Table 3.2. In this research, the SSM safety setup is based on the methodology of ellipsoidal approximation for a 6DOF robot as a 3-DOF ellipsoids in space. The validation of this approach using a digital-twin simulation setup and the implementation of dynamic SSM using on-robot ToF sensor array is presented in the following chapter.

Chapter 4

Dynamic SSM using on-robot Time-of-Flight Laser Ranging Sensor Arrays

The core contribution of this dissertation is presented in this chapter. The formulation and implementation of dynamic SSM using on-robot ToF sensor arrays is discussed. There are two parts to this chapter. In the first part, a validation of a dynamic SSM safety measure with simulated ToF sensor arrays in a digital-twin is detailed. The second part formulates the dynamic SSM for on-robot sensor arrays and presents the implementation and results with real ToF sensor array hardware placed at the centers of UR10 robot links.

4.1 Validation of Dynamic SSM with simulated ToF Sensor Arrays

In this section, a *digital-twin* simulation setup with simulated ToF ranging sensor placed on a UR10 robot performing a a standard pick and place material handling task with a human avatar is used to analyze the behavior and viability of a dynamic SSM safety measures. The block diagram of the digital-twin setup is shown in Figure 4.1. Speed and separation monitoring approaches based on human-robot separation distance, and relative speeds using the information from the ToF sensor arrays is described. These ToF-based dynamic SSM safety approaches are compared with the conventionally used 2D LiDAR-based static SSM approaches. These safety approaches are evaluated based on the human safety, robot performance, and productivity of the task.

In Speed and Separation Monitoring, based on the minimum separation distance and relative velocities between the robot and the human, the robot stops or slows its motion to avoid a collision. The conventional setups of SSM in industry use 2D scanning LiDARs to measure the distance between robot and human. These setups have fixed 2D safety zones around the robot demarcated using fixed distance thresholds. This form of perception can

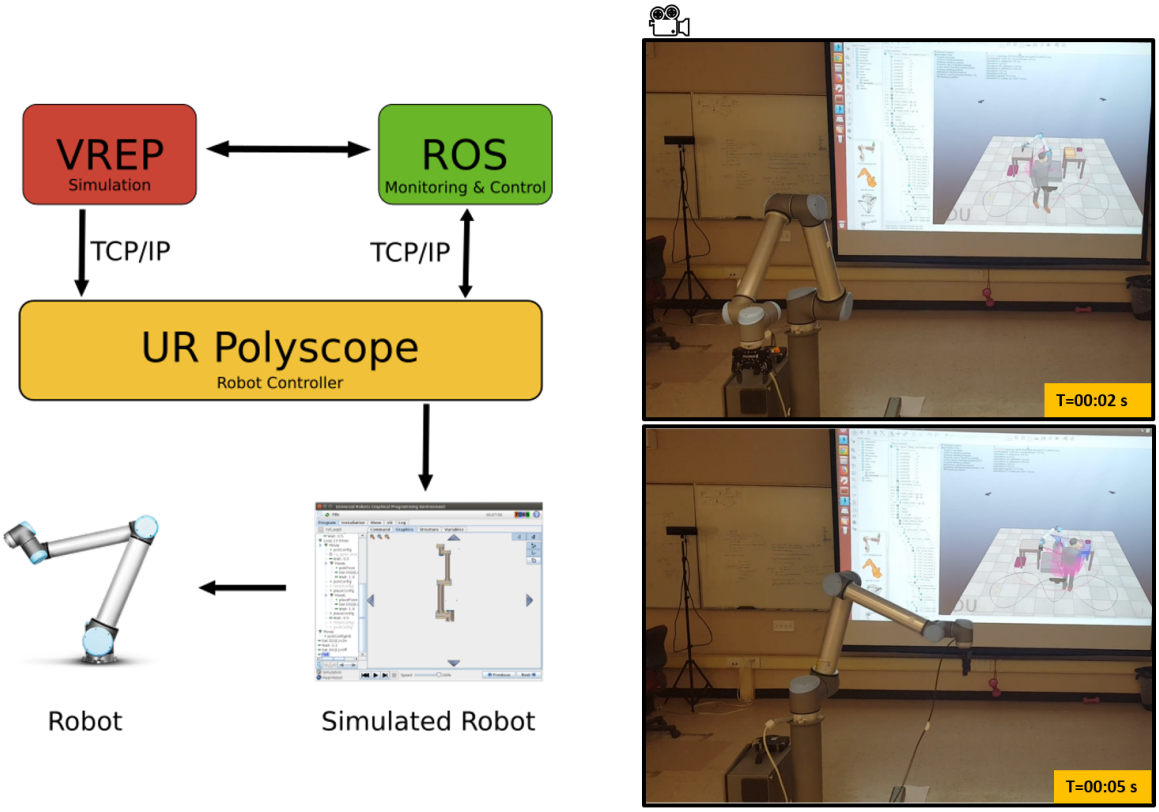


Figure 4.1: A block diagram of the digital-twin setup, also known as simulation-in-the-loop for the UR10 robot and the video frames of this setup, where the UR10 robot in the physical world 'hallucinates' the moving human and the environment represented in the V-REP 3D simulation. Video Link: https://youtu.be/39t_xQS6KSE

be termed as 'static awareness' of the robot, and the safety configuration is termed as 'Static SSM.' Such setups are inefficient in cases where the human overlaps with robot operating workspace more often, resulting in frequent robot halts, thereby negatively impacting the overall productivity.

A more flexible and dynamic implementation of safety measures that optimizes the productivity of a task while ensuring human safety is needed. Unlike the static/fixed demarcated 2D safety zones, a dynamic sensor setup that changes the safety zones based on the robot pose and motion is presented in this dissertation. An SSM safety configuration implemented based on a dynamic sensor setup would result in 'Dynamic SSM.' In Figure 4.2, the zones of detection for a 2D LiDAR and ToF ranging sensors mounted on the robot is shown.

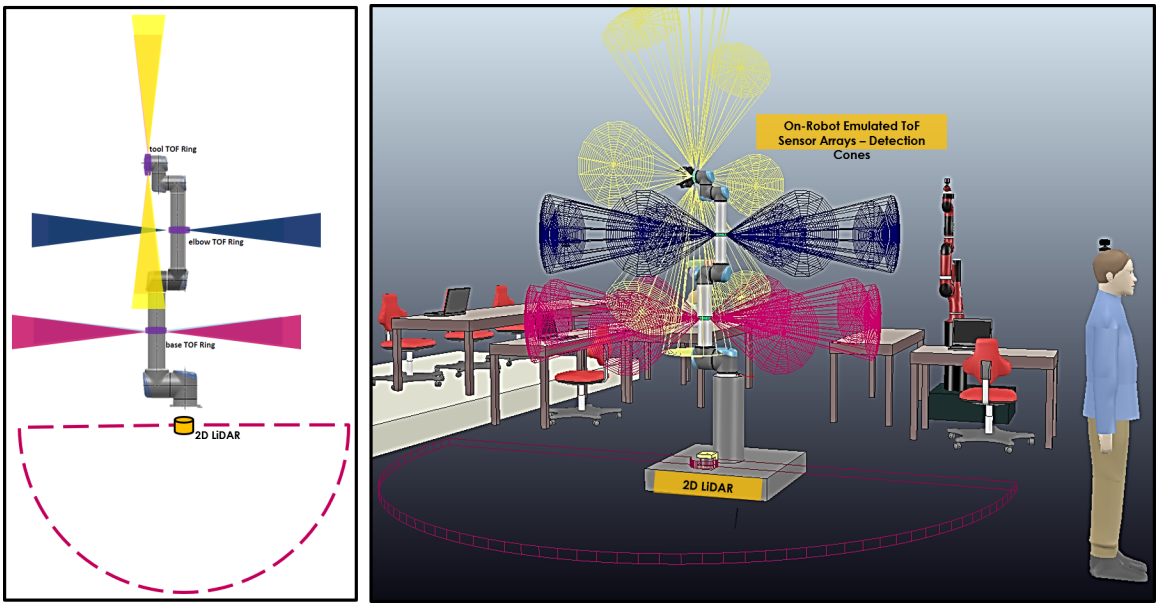


Figure 4.2: The simulated detection area of the 8 Node ToF Sensor arrays placed at the centers of the robot links; and the 2D LiDAR scanner detection zone.

In a human-robot collaboration setup, the safety or danger of the robot to the human agent needs to be quantified. Many researchers have proposed safety measures such as safety indices [57, 71, 82] that depend on the relative distance between the human and the robot, potential impact force at the time of collision, momentum of the robot end-effector or relative human-robot velocities.

In [120], the authors define the impact potential between any-part of the robot and a static object. The work in [57, 95, 110] defines safety index-based on effective robot inertia and human-robot relative velocities and distance. The distance is measured between a predefined key-point on the robot and a point on the human agent or object. In the case when an entire robot arm is considered, multiple key-points on the robot are assigned. For each key-point on the robot link, a search for the nearest point on the human agent or object is required. Thus, the practical implementation of such collision control strategies require extra computation to search for the closest point.

For on-robot sensors, the work done in [19, 71] used kinetostatic fields, which are a function of the robot pose and the velocity of the key-points on the robot. An integral of the fields calculated on the points of the robot link provides the danger index for the robot link. The danger index can be calculated for all the points in the workspace. Thus, in

addition to the computation of the kinetostatic danger field, more computation is required to search for the point of interest on the agent.

The work presented in [82] addresses the high computational cost, by defining safety indices using ellipsoidal geometric approximations of the robot links and human agent body parts. The author in [82] illustrates two safety indices, distance safety index (DSI) and momentum safety index (MSI), based on human-robot distance and linear momentum of the robot links to quantify safety. This approach is computationally less expensive, as instead of defining key-points on the robot links and searching for the closest point on the human agent, ellipsoid approximations for both the robot-links and human or parts of the human are used.

Based on the work in [82], a dynamic SSM safety approach using on-robot simulated ToF sensor arrays is implemented by defining the DSI and MSI safety indices. This is discussed in the following Section.

4.1.1 Design of Safety Indices for Dynamic SSM

In [82], the authors define the relative human-robot distance using the ellipsoidal approximations of the human and robot links. An example of a 2D planer robot is used to explain the DSI and MSI safety indices as defined in [82], as shown in Figure 4.3.

4.1.1.1 Distance Safety Index (DSI)

The relative human-robot distance is directly related to the spatial overlap of human and robot, and the safety of the setup. The shorter the distance, higher the chance of a human-robot collision. Hence, the DSI has been defined as the reciprocal of the human-robot distance.

A planer two-link robot example is used for defining the safety indices using ellipsoid approximation, as shown in Figure 4.3(a). Here, the ellipsoid approximation is centered at P_{l_j} corresponding to the j-th robot link and an ellipsoid is centered at P_{o_i} corresponding to the i-th ellipsoid approximation of the agent/object. The position vectors with respect to the universal/world fixed frame for the centers of the robot link and agent ellipsoid are $\boldsymbol{r}_{\boldsymbol{c}_j}$ and $\boldsymbol{r}_{\boldsymbol{o}_i}$, respectively. The shortest distance d_{ji} used for defining DSI is calculated from the center robot link P_{l_j} to the closest point on the agent/object i-th ellipsoid. The search for the

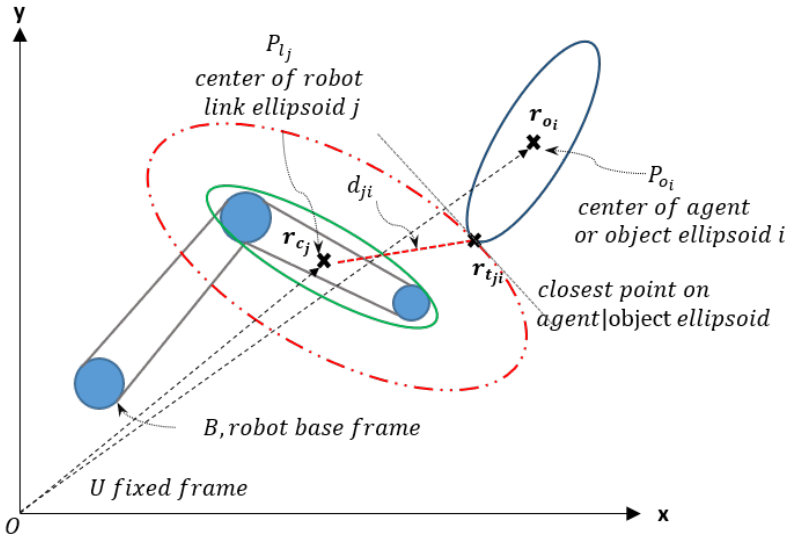


Figure 4.3: A planer ellipsoid example for defining Safety Index.

closest point is less computationally expensive by defining the d_{ji} in ellipsoid coordinates with origin at the robot link center $\boldsymbol{r}_{c_j}$. This is achieved by defining a new ellipsoid centered at the robot link center ($\boldsymbol{r}_{c_j}$) and tangential to the agent/object ellipsoid. The magnitude, specifically the L2 norm of the displacement between the center of the robot link $\boldsymbol{r}_{c_j}$ and the resulting tangential point $\boldsymbol{r}_{t_j}$ is defined as the minimum/shortest distance $\|\boldsymbol{r}_{c_j} - \boldsymbol{r}_{t_{ji}}\|_2 = d_{ji}$.

Hence, the danger for i-th agent/object ellipsoid posed by the j-th robot link for a given a robot pose (joint angles $\boldsymbol{q}$) is quantified by defining DSI for the pair of (j, i) as:

$$dsi_{ji}(\boldsymbol{q}) = \frac{1}{(d_{ji})^2} \tag{4.1}$$

4.1.1.2 Momentum Safety Index (MSI)

The robots moving at higher velocity would require a longer time and longer distance to stop the motion after the stop command is issued. Hence the momentum of the robot links is a factor that is considered in defining the momentum safety index (MSI). The linear momentum of the robot links is computed from the center of the robot link ellipsoid ($\boldsymbol{r}_{c_j}$) towards the direction of the agent ($\boldsymbol{r}_{o_i}$). This is calculated by taking the scalar projection of the linear momentum vector onto the relative position vector from the center of j-th robot

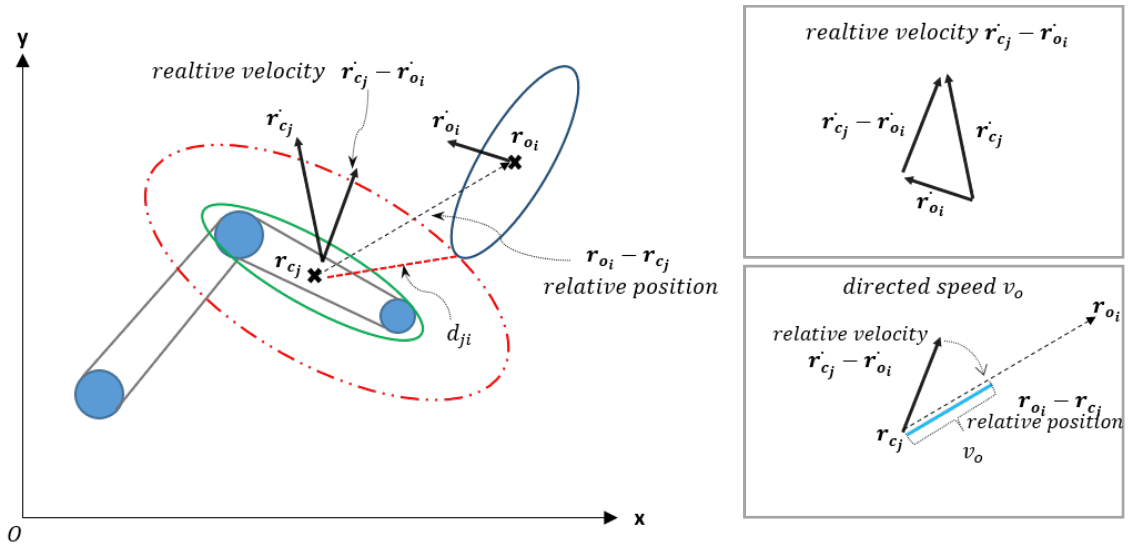


Figure 4.4: A planer ellipsoid example showing the scalar projection of the relative velocity vector onto the relative position vector.

link ellipsoid to the center of i-th agent/object ellipsoid ($\boldsymbol{r}_{o_i} - \boldsymbol{r}_{c_j}$) [82]. This projection p_{ji} is defined as,

$$
\begin{aligned}
p_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= M_{l_j} \underbrace{\left(\dot{\boldsymbol{r}}_{c_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\boldsymbol{r}}_{o_i}\right)^T \left(\frac{\boldsymbol{r}_{o_i} - \boldsymbol{r}_{c_j}(\boldsymbol{q})}{\|\boldsymbol{r}_{o_i} - \boldsymbol{r}_{c_j}(\boldsymbol{q})\|_2}\right)}_{\text{directed speed } v_{o_{ji}}} \\
&= M_{l_j}\, v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}})
\end{aligned}
\tag{4.2}
$$

where M_{l_j} is the mass and $\dot{\boldsymbol{r}}_{c_j}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the linear velocity vector of the j-th robot link ellipsoid. The linear velocity $\dot{\boldsymbol{r}}_{c_j}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is a function of the robot joint positions and velocity. $\dot{\boldsymbol{r}}_{o_i}$ is the linear velocity of the i-th agent/object ellipsoid. The directed speed $v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the projected scalar component of the relative velocity vector $\left(\dot{\boldsymbol{r}}_{c_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\boldsymbol{r}}_{o_i}\right)$ on to the relative position vector ($\boldsymbol{r}_{o_i} - \boldsymbol{r}_{c_j}(\boldsymbol{q})$), as shown in Figure 4.4. The physical meaning of directed speed $v_{o_{ji}}$ can be interpreted as the magnitude of the speed of the robot link moving towards the agent/object.

The MSI for the pair of (j, i) based on p_{ji} is defined in [82] as,

$$msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{(d_{ji}^2)}\Big[\max\ \Big\{\ p_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}),\ 0\Big\}\Big]^2 \tag{4.3}$$

In other words, $msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ can be rewritten as

$$msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max\ \left\{\text{sign}(p_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\ \left(\frac{p_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{d_{ji}}\right)^2,\ 0\right\} \tag{4.4}$$

Here, the mass of the links are not considered and assumed as constant for all links, for validation purposes. Hence in the Eq.(4.5), the scalar momentum $p_{ji} = M_{l_j} \cdot v_{o_{ji}}$ for all links having unit mass, can be written with the directed speed $vo_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ as

$$msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max\ \left\{\text{sign}(v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\ \left(\frac{v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{d_{ji}}\right)^2,\ 0\right\} \tag{4.5}$$

The maximization in Eq.(4.3) indicates that only the case when the robot link is moving towards the agent is considered in MSI. A negative directed speed $v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ indicates the robot link moving away from the agent/object and is considered as a safe situation. The denominator (d_{ji}^2) is used to differentiate between (j, i) pairs in cases when the directed speeds are the same, but the relative positions of the agent ellipsoids are different. A robot link moving at high velocity towards an agent that is far away should be considered less dangerous. Another interpretation of the MSI is that it scales the DSI for the robot link based on the calculated directed speeds. Thus, using Eq.(4.1) the Eq.(4.5) can be rewritten as

$$msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max\ \left\{\text{sign}(v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\ \left(v_{o_{ji}}(\boldsymbol{q}, \dot{\boldsymbol{q}})^2\ dsi_{ji}(\boldsymbol{q})\right),\ 0\right\} \tag{4.6}$$

4.1.1.3 Overall Safety Index

The safety indices DSI and MSI indicate that smaller their value, the higher the safety. An overall safety index $si(\boldsymbol{q}, \dot{\boldsymbol{q}})$ of the robot is defined by first combining $dsi_{ji}(\boldsymbol{q})$ and $msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ for all (j, i) pairs, and selecting the maximum value. This ensures the capture of least safe (maximum danger) scenario indicated by DSI and MSI for each link. The combination chosen for the validation for the overall safety index is a linear combination of $dsi_{ji}(\boldsymbol{q})$ and $msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, and then selecting the maximum value.

The overall safety index $si(\boldsymbol{q}, \dot{\boldsymbol{q}})$ can be defined as

$$si(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max_{j,i} \Big\{ k_{dsi} \cdot dsi_{ji}(\boldsymbol{q}) + k_{msi} \cdot msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \Big\} \tag{4.7}$$

where the weights k_{dsi} and k_{msi} are the inverse of the maximum acceptable DSI (dsi_{max}) and MSI (msi_{max}), respectively, and are chosen for normalizing DSI and MSI. The dsi_{max} can be defined by choosing the acceptable/allowable minimum distance or the minimum distance threshold $d_{min_{thresh}}$ as,

$$\begin{aligned} k_{dsi} &= \frac{1}{dsi_{max}} \\ &= d^2_{min_{thresh}} \end{aligned} \tag{4.8}$$

Similarly, msi_{max} can be defined by choosing the maximum acceptable directed speed $v_{max_{thresh}}$ and the acceptable minimum distance $d_{min_{thresh}}$ as,

$$\begin{aligned} k_{msi} &= \frac{1}{msi_{max}} \\ &= \left(\frac{d_{min_{thresh}}}{v_{max_{thresh}}} \right)^2 \end{aligned} \tag{4.9}$$

The normalized $\widetilde{dsi_{ji}(\boldsymbol{q})}$ and $\widetilde{msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})}$ can be defined as

$$\widetilde{dsi_{ji}(\boldsymbol{q})} = \begin{cases} 1, & \text{if } dsi_{ji}(\boldsymbol{q}) > dsi_{max} \\ \frac{dsi_{ji}(\boldsymbol{q})}{dsi_{max}}, & \text{otherwise} \end{cases} \tag{4.10}$$

and

$$\widetilde{msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})} = \begin{cases} 1, & \text{if } msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}}) > msi_{max} \\ \frac{msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{msi_{max}}, & \text{otherwise} \end{cases} \tag{4.11}$$

Hence the overall safety index can be rewritten as

$$si(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max_{j,i} \Big\{ \underbrace{\widetilde{dsi_{ji}(\boldsymbol{q})}}_{\leq 1} + \underbrace{\widetilde{msi_{ji}(\boldsymbol{q}, \dot{\boldsymbol{q}})}}_{\leq 1} \Big\} \tag{4.12}$$

Thus, $si(\boldsymbol{q}, \dot{\boldsymbol{q}}) \leq 2$, with the worst-case scenario (least safe or greatest danger) value of the overall safety index $si(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 2$.

The overall safety index $si(\boldsymbol{q}, \dot{\boldsymbol{q}})$ can be used to determine the robot action for the implementation of dynamic SSM. In the next section, the overall safety index is formulated using the information available from the on-robot ToF sensor array setup.

4.1.2 Safety Indices using ToF Sensor Array

In this section, the calculation of overall safety index for ToF sensor arrays placed on the robot links is presented. The ellipsoid approximation of the 6-DOF UR10 robot as a 3-DOF robot is shown in Figure 4.5. The notations for the robot link ellipsoids are l_j for the j-th robot links, where $j = \{1\ (base),\ 2\ (elbow),\ 3\ (tool)\}$. A ToF sensor array S_j consists of eight ToF sensor nodes S_{ji} (where $i = \{1, 2, \ldots, 8\}$) mounted as a ring around the j-th robot link. The geometric transformation chain for the observed point $\boldsymbol{P_{o_j}}$ from sensor node S_{ji} is shown in Figure 4.5. The transformation of the sensor array on robot link l_j is a function of the robot joint angles ($\boldsymbol{q}$) . The robot base frame B is static with respect to the fixed universal frame U as the robot base does not move. The sensor node S_{ji} is also fixed with respect to the robot link l_j.

In order to calculate the overall safety index ($si(\boldsymbol{q}, \dot{\boldsymbol{q}})$), the distance safety index (dsi_j) and the momentum safety index (msi_j) for the all the robot link ellipsoids (*base*, *elbow* and *tool*) are required. The derivations of dsi_j and msi_j using the distance observations of the ToF sensor arrays are presented in the following sections.

4.1.2.1 Defining DSI using ToF Sensor Array

In order to calculate DSI according to Eq.(4.1), the minimum distance $d_{j\,min}$ between the j-th robot link center point P_{l_j} and the closest point on the agent P_{o_j} needs to be calculated using the distance observations from all the ToF sensor nodes ($S_{ji},\ i = 1, 2, \ldots, 8$) in the sensor array S_j placed on the robot link. A point on the agent P_{o_j} as observed by a ToF sensor node S_{ji} can be written using geometrical transformation as,

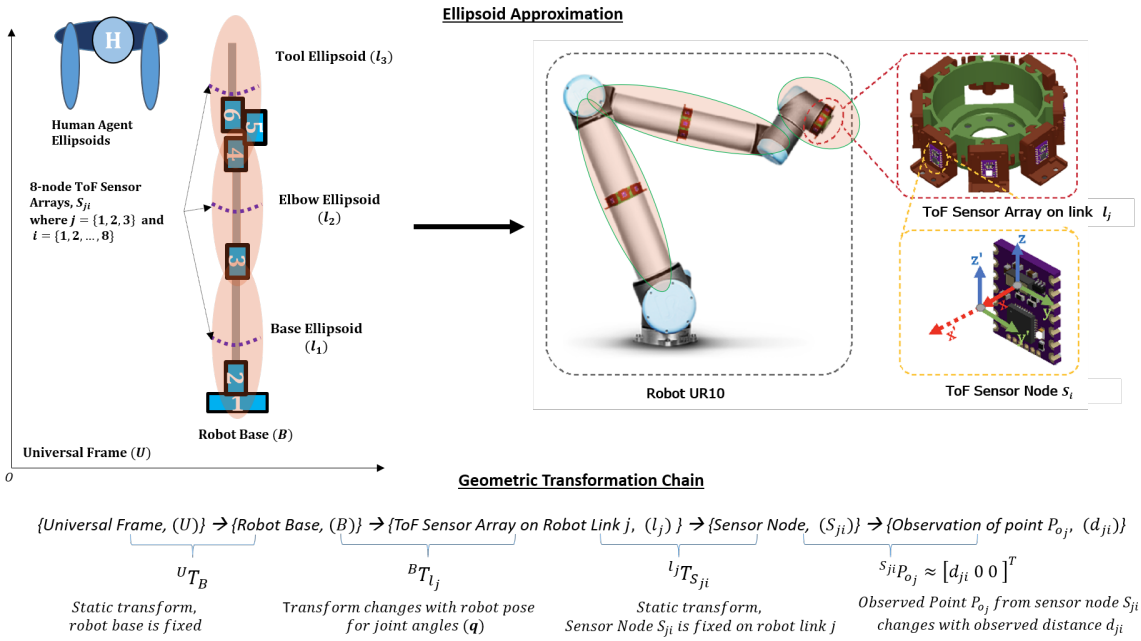


Figure 4.5: The ellipsoid approximation of the ToF sensor array setup and the geometric transformation chain for the observed point $\boldsymbol{P_{o_j}}$ from sensor node S_{ji} of the sensor array placed on robot link l_j.

$$\begin{aligned} \boldsymbol{P_{o_j}} &= {}^{U}\boldsymbol{T_B}\ {}^{B}\boldsymbol{T_{l_j}}\ {}^{l_j}\boldsymbol{T_{S_{ji}}}\ {}^{S_{ji}}\boldsymbol{P_{o_j}} \\ &= {}^{U}\boldsymbol{T_{l_j}}\ {}^{l_j}\boldsymbol{T_{S_{ji}}} \begin{bmatrix} x_{ji} \\ y_{ji} \\ z_{ji} \\ 1 \end{bmatrix} \end{aligned} \tag{4.13}$$

where ${}^{S_{ji}}P_{o_j} = \begin{bmatrix} x_{ji}\ y_{ji}\ z_{ji} \end{bmatrix}^T$ is the position of the point on the agent with reference from the sensor node S_{ji}. However, the observation reported by a ToF sensor node S_{ji} is a single distance value d_{ji} of the point in its field-of-view. Hence, an approximation is made to represent P_{o_j} only in the direction of ranging measurement, which is the X-axis as ${}^{S_{ji}}P_{o_j} \approx \begin{bmatrix} d_{ji}\ 0\ 0 \end{bmatrix}^T$, as shown in Figure 4.6. The position vector of the perceived or approximated point $\boldsymbol{r_{o_j}}$ can be written as

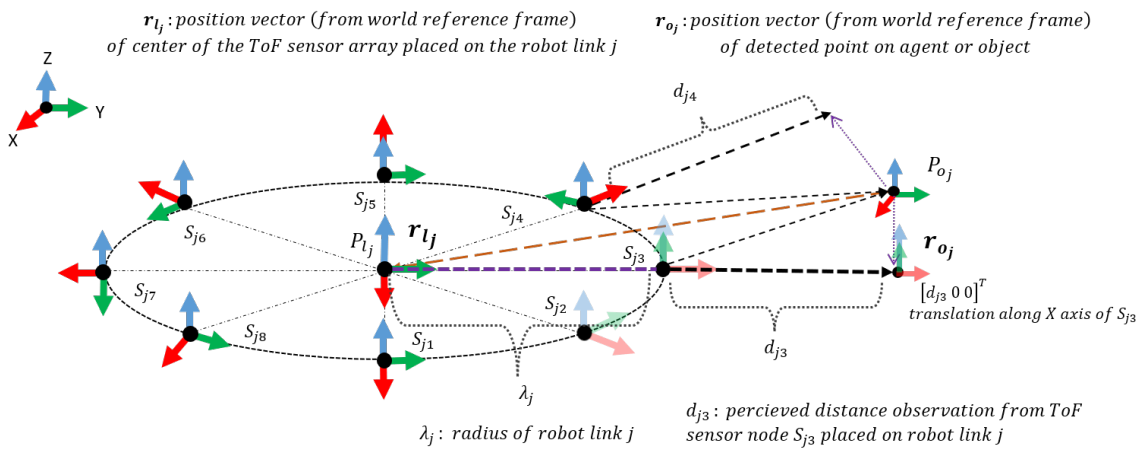


Figure 4.6: This diagram shows the reference frames of eight ToF sensor nodes(S_{ji}) in the ToF sensor array (S_j) placed as a ring at the center of the j-th robot link (l_j). A point p_{o_j} is detected by the sensor nodes S_{j3}.

$$\boldsymbol{r_{o_j}} = {}^{U}\boldsymbol{T}_{l_j}\ {}^{l_j}\boldsymbol{T}_{S_{ji}} \begin{bmatrix} d_{ji} \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{4.14}$$

The ToF sensor nodes (S_{ji}) are fixed with reference to the j-th robot link frame (l_j) and the transformation of the sensor array on robot link frame (l_j) is a function of the robot joint angles ($\boldsymbol{q}$). For a given radius λ_j of the ToF sensor array ring S_j on the j-th robot link, $r_{o_j}(\boldsymbol{q})$ can be rewritten as,

$$\begin{aligned} \boldsymbol{r_{o_j}}(\boldsymbol{q}) &= {}^{U}\boldsymbol{T}_{l_j}(\boldsymbol{q}) \left[\begin{array}{ccc|c} & {}^{l_j}\boldsymbol{Rot}_{S_{ji}} & & \begin{matrix}\lambda_j\\0\\0\end{matrix} \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \begin{bmatrix} d_{ji} \\ 0 \\ 0 \\ 1 \end{bmatrix} \\ &= \underbrace{{}^{U}\boldsymbol{T}_{l_j}(\boldsymbol{q})}_{r_{l_j}} \left[\begin{array}{ccc|c} & {}^{l_j}\boldsymbol{Rot}_{S_{ji}} & & \begin{matrix}0\\0\\0\end{matrix} \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \begin{bmatrix} \lambda_j + d_{ji} \\ 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned} \tag{4.15}$$

where ${}^{l_j}\boldsymbol{Rot}_{S_{ji}}$ is the rotation transformation from the robot link frame (l_j) to the ToF sensor node frame (S_{ji}). The position vector of the point on the agent from the j-th robot link from l_j depends only on the distance observation d_{ji} reported by i-th sensor node as the orientation ${}^{l_j}\boldsymbol{Rot}_{S_{ji}}$ and the radius λ_j are constant with respect to the robot center link frame l_j. Hence, the minimum/shortest distance ($d_{j\,min} = \|\boldsymbol{r}_{o_j} - \boldsymbol{r}_{l_j}\|_2$) of the closest point on the agent/object observed by the ToF sensor array (S_j) from the center of the j-th robot link (l_j) can be defined as

$$\begin{aligned} d_{j\,min} &= \lambda_j + \min\left\{d_{j1}, d_{j2}, \ldots, d_{j8}\right\} \\ &= \lambda_j + \min_i\left\{d_{ji}\right\} \end{aligned} \tag{4.16a}$$

$$k_j = \underset{i}{\operatorname{argmin}}\left\{d_{ji}\right\} \tag{4.16b}$$

where k_j-th ToF sensor node in the array that reported the minimum distance observation of the closest point on the agent from the j-th robot link. The position and orientation of the closest detected point in reference to the j-th robot link center frame l_j are $\left[d_{j\,min}\ 0\ 0\right]$ and ${}^{l_j}Rot_{S_{jk_j}}$, respectively. Thus, the DSI for the j-th robot link using Eq.(4.1) can be written as

$$dsi_j = \left(\frac{1}{d_{j\,min}}\right)^2 \tag{4.17}$$

The MSI using the information from the ToF sensor array is defined in the following section.

4.1.2.2 Defining MSI using ToF Sensor Array

For MSI, the directed speed v_{o_j} of the j-th robot link moving towards the human-agent is required (refer Eq.(4.2) and Eq.(4.5)). The velocity vector ($\dot{\boldsymbol{r}}_{o_j}$) of closest detected point on the human agent/object ellipsoid from the j-th robot link can be defined as

$$\dot{\boldsymbol{r}}_{o_j} = \left(\frac{\boldsymbol{r}_{o_j}(t) - \boldsymbol{r}_{o_j}(t - \Delta t)}{\Delta t}\right) \tag{4.18}$$

where $\boldsymbol{r}_{o_j}(t)$ and $\boldsymbol{r}_{o_j}(t-\Delta t)$ are the position vector at time (t) and $(t-\Delta t)$, respectively, for a give (Δt) simulation time step. The position vector $\boldsymbol{r}_{l_j}$ and velocity vector $\dot{\boldsymbol{r}}_{l_j}$ of the j-th robot link ellipsoid are a function of the robot joint pose and joint velocities $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and are reported in the simulation.

Hence, based on Eq.(4.2) and Eq.(4.5), the directed speed v_{o_j} and msi_j of the j-th robot link can be written as

$$v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \left(\dot{\boldsymbol{r}}_{l_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\boldsymbol{r}}_{o_j}\right)^T \left(\frac{\boldsymbol{r}_{o_j} - \boldsymbol{r}_{l_j}(\boldsymbol{q})}{\|\boldsymbol{r}_{o_j} - \boldsymbol{r}_{l_j}(\boldsymbol{q})\|_2}\right) \tag{4.19a}$$

$$msi_j(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \max\left\{\text{sign}(v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\left(\frac{v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{d_{j\,min}}\right)^2, 0\right\} \tag{4.19b}$$

The overall safety index $si(\boldsymbol{q}, \dot{\boldsymbol{q}})$ for the robot using ToF sensor arrays can be written again using Eq.(4.7)- Eq.(4.12) as

$$\begin{aligned} si(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= \max_j\Big\{\underbrace{\widetilde{dsi_j}}_{\leq 1} + \underbrace{\widetilde{msi_j(\boldsymbol{q}, \dot{\boldsymbol{q}})}}_{\leq 1}\Big\} \\ &= \max_j\left\{\frac{dsi_j}{dsi_{max}} + \frac{msi_j(\boldsymbol{q}, \dot{\boldsymbol{q}})}{msi_{max}}\right\} \\ &= \max_j\left\{\frac{d^2_{min_{thresh}}}{d^2_{j\,min}} + \frac{d^2_{min_{thresh}}}{v^2_{max_{thresh}}}\left(\max\left\{\text{sign}(v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\left(\frac{v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{d_{j\,min}}\right)^2, 0\right\}\right)\right\} \\ &= \max_j\left\{\left(\frac{d_{min_{thresh}}}{d_{j\,min}}\right)^2 + \max\left\{\text{sign}(v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}}))\left(\frac{v_{o_j}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{v_{max_{thresh}}}\frac{d_{min_{thresh}}}{d_{j\,min}}\right)^2, 0\right\}\right\} \end{aligned} \tag{4.20}$$

In the next section, the safety approaches used for the implementation of dynamic SSM using the safety indices with simulated ToF sensor arrays are discussed.

4.1.2.3 Safety Approaches using ToF Sensor Array

The safety approaches to determine the robot actions using the distance measurements and safety indices are discussed in this section. Different variations of dynamic SSM safety approach based on simulated ToF sensor setup using the safety indices and the distance observations is presented. For comparison of dynamic and static SSM, a 2D LiDAR-based

static SSM is also implemented.

A. Robot Actions

An SSM safety approach is a pre-collision control strategy of controlling the overall robot motion by limiting the speed of the robot (refer Section 3.1.4.1). In this approach, the trajectory of the robot motion performing a task remains unchained. The robot actions can be defined as a robot operating at three different speed levels: *stop, reduced (slow) and normal*. The UR10 robot controller provides an option to limit the overall operating speed by a scalar quantity called the 'speed fraction' [11, 59].

Let this speed fraction scalar be ρ, and the robot action or the desired mode of operation for the robot be denoted as $\psi \in \{\psi_{stop} = 0, \psi_{reduced} = 1, \psi_{normal} = 2\}$. Given the robot joint pose and joint angular velocities $(\boldsymbol{q}, \dot{\boldsymbol{q}})$, the linear velocity of the Tool Control Point (TCP) or the end-effector $(\dot{\boldsymbol{r}}_e)$ can be determined using the Jacobian $(\boldsymbol{J}_e)$ as

$$\dot{\boldsymbol{r}}_e = \boldsymbol{J}_e \dot{\boldsymbol{q}} \tag{4.21}$$

Thus, the Jacobian $(\boldsymbol{J}_e)$ of the robot can be computed using the forward propagation of joint angular velocities $((\dot{\boldsymbol{q}}))$ to determine the TCP velocity $(\dot{\boldsymbol{r}}_e)$. The effect of reducing/limiting the joint angular velocities $(\dot{\boldsymbol{q}})$ by the scalar ρ on the TCP velocity $(\dot{\boldsymbol{r}}_e)$ can be written as

$$\rho\, \dot{\boldsymbol{r}}_e = \boldsymbol{J}_e(\rho\, \dot{\boldsymbol{q}}) \tag{4.22}$$

$$\rho\, \dot{\boldsymbol{r}}_e = \rho\, (\boldsymbol{J}_e \dot{\boldsymbol{q}}) \tag{4.23}$$

This shows that by scaling all the joint angular velocities, the end-effector velocity changes proportionally without affecting the robot trajectory.

In this research, the scalar speed fraction ρ of the operational robot speed in relation to the robot actions is defined as

$$\text{where } \rho = \begin{cases} 0, & \text{if } \psi = \psi_{stop} \\ 0.5, & \text{if } \psi = \psi_{reduced} \\ 1, & \text{if } \psi = \psi_{normal} \end{cases} \tag{4.24}$$

The physical meaning of ρ for the robot action (or robot operational mode) ψ_{normal} is the robot moves at the maximum designated speed for performing the task, for $\psi_{reduced}$ the robot moves at half the maximum speed and the robot does not move for the ψ_{stop}. In order to prevent the robot motion to change drastically, in addition to the speed fraction (ρ), a rate of change of the speed fraction ($\Delta\rho$) at each simulation time step (Δt) is also defined. The robot action transition to achieve the target ρ_{target} is changed gradually (increased or decreased) at every simulation time step (Δt) as $\rho_{target} = \rho(t) \pm \Delta\rho$, where $\rho(t)$ is the current value of speed fraction ρ at time (t). Therefore, each robot action/operation mode ($\psi \leftarrow (\rho_{target}, \Delta\rho)$) can be defined by the pair of speed fraction scalar (ρ_{target}) and the rate of change ($\Delta\rho$) at each simulation time step.

In many conventional SSM safety approaches, the reduced robot operation mode ($\psi_{reduced}$) is not considered. In order to differentiate the SSM safety approaches with three robot actions/modes of operation (ψ_{normal}, $\psi_{reduced}$, and ψ_{stop}), they are termed as *Tri-Modal SSM* or *Tri-SSM* for short. In this research, the dynamic SSM safety approaches implemented are Tri-Modal SSM. However, for validation and comparison purposes in this experiment two/dual operation mode ($\psi_{normal}, \psi_{stop}$) SSM have been considered.

The following sections present the algorithms for different safety approaches used for the evaluation of dynamic SSM using ToF sensor arrays and static SSM using 2D scanning LiDAR.

B. 2D LiDAR-based Static SSM

In this setup, the simulated 2D LiDAR is placed at the base of the robot near the floor, as shown previously in Figure 4.2. Static safety zones using the 2D LiDAR can be determined by defining safety distance thresholds $d_{2D_{stop}}$ and $d_{2D_{reduce}}$, as shown in Figure 4.7. For a give 2D LiDAR distance measurement ($d_{2D}(t)$) at time (t) of an agent in the workspace, the Tri-SSM safety algorithm shown in Algorithm 1 is used to determine the robot action/operation mode.

As seen in Algorithm 1(*Lines [1-2]*), the robot action is only taken when the 2D LiDAR detects the agent. A robot action ψ is taken by setting a corresponding speed fraction ρ and the rate of change of the speed fraction $\Delta\rho$. In case of Tri-SSM the target speed fraction scalars ($\rho_{normal} = 1$, $\rho_{reduce} = 0.5$, $\rho_{stop} = 0$) and speed fraction rates ($\Delta\rho_{normal}$, $\Delta\rho_{reduce}$, $\Delta\rho_{stop}$) are predefined. For implementing the dual-mode SSM where the $\psi_{reduced}$ is not a

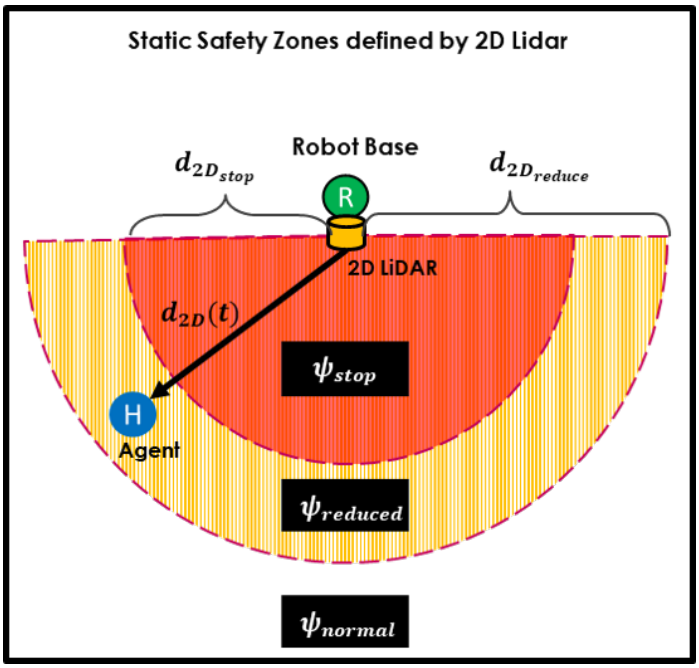


Figure 4.7: Static safety zones defined using 2D LiDAR by the protective/critical safety distance threshold $d_{2D_{stop}}$ and the reduced safety distance threshold $d_{2D_{reduce}}$. The distance measurement by the 2D LiDAR of an agent in the robot workspace at time (t) is given as $d_{2D}(t)$.

robot action, the Algorithm 1 can be modified by omitting the conditional statements on *Lines [8-11]*.

Algorithm 1: Static Tri-SSM Algorithm using 2D LiDAR sensor

Data: safety distance thresholds ($d_{2D_{stop}}$, $d_{2D_{reduce}}$)
speed fraction ($\rho_{normal} = 1$, $\rho_{reduce} = 0.5$, $\rho_{stop} = 0$)
speed fraction rates ($\Delta\rho_{normal}$, $\Delta\rho_{reduce}$, $\Delta\rho_{stop}$)
Input : distance measured by 2D Lidar $d_{2D}(t)$
Output: robot action ψ

```
1  if d_2D(t) is out of range then
       /* do nothing                                          */
2      robot action ψ ← ψ_normal
3  else
4      if d_2D(t) ≤ d_2D_stop then
           /* stop the robot                                  */
5          setSpeedFraction(ρ_stop)
6          setSpeedFractionRate(Δρ_stop)
7          robot action ψ ← ψ_stop
8      else if d_2D_stop < d_2D(t) ≤ d_2D_reduce then
           /* reduce the robot speed                          */
9          setSpeedFraction(ρ_reduce)
10         setSpeedFractionRate(Δρ_reduce)
11         robot action ψ ← ψ_stop
12     else
           /* robot moves normally                            */
13         setSpeedFraction(ρ_normal)
14         setSpeedFractionRate(Δρ_normal)
15         robot action ψ ← ψ_normal
16     end
17 end
```

The ToF sensor-based dynamic SSM approaches are presented in the following sections.

C. ToF Binary Detection based Dynamic SSM

A binary detection based dynamic SSM safety approach is implemented to validate the basic safety setup of Dynamic SSM using on-robot ToF sensor arrays. This is a dual-mode dynamic SSM approach that results in two robot actions ψ_{stop} or ψ_{normal}. This is considered as a dynamic SSM setup as the sensing zones of the ToF sensor placed on the robot changes with the robot pose. A cone-shaped proximity sensor with a field-of-view angle of 25° degrees and a maximum height or detection range of $1.3m$ is simulated as a ToF sensor node in V-REP simulation (see Figure 4.8 and Figure 4.2). A distance measurement observation o_{ji} from the proximity sensor corresponding to the simulated ToF sensor node S_{ji} is only available if the agent/object is within the detection range. Thus the distance measurement d_{ji} of the ToF sensor node S_{ji} is defined as

$$d_{ji} = \begin{cases} 1.3m, & \text{if no detection by the proximity sensor} \\ o_{ji}, & \text{otherwise} \end{cases}$$

In this safety setup, the robot stops if any of the ToF sensor nodes S_{ji}, in a ToF sensor array S_j, placed on the j-th robot link detects a point on the agent. In other words, if the human agent is within the sensing range of the ToF sensor arrays, the robot stops, otherwise continues to move normally. This is a highly conservative safety setup that results into a high number of robot stops. This is shown in Algorithm 2. At any given time (t) the minimum/shortest distance of a point on the agent for ToF sensor array S_j placed in j-th robot link (where $j = \{1(base), 2(elbow), 3tool\}$), $d_{j\,min}(t)$ can be calculated using Eq.(4.16a). The reported minimum/shortest distance measurements ($d_{j\,min}(t)$, $\forall\ j$) for all the ToF sensor arrays are used as input for the Algorithm 2. At any given time (t), the overall minimum distance $d_{min}(t)$ and the j^*-th link that reported the closest point can be defined as

$$d_{min}(t) = \min_j \{d_{j\,min}(t)\} \tag{4.25a}$$

$$j^* = \operatorname*{argmin}_j \{d_{j\,min}(t)\} \tag{4.25b}$$

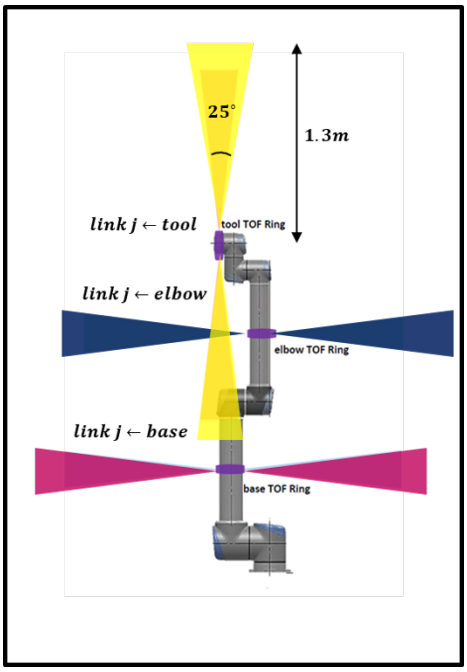


Figure 4.8: A diagram showing the detection region of a cone-shaped proximity sensor with a field-of-view angle of 25° degrees and a maximum height or detection range of 1.3m used to simulate the ToF sensor node.

Algorithm 2: Binary Detection SSM Algorithm using ToF Sensor Arrays

Data: speed fraction ($\rho_{normal} = 1, \rho_{stop} = 0$)
speed fraction rates ($\Delta\rho_{normal}, \Delta\rho_{stop}$)
Input : the minimum/shortest distance $d_{j\,min}(t),\ \forall\ j = \{base, elbow, tool\}$
Output: robot action ψ

$d_{min}(t) \leftarrow \min_{j}\{d_{j\,min}\}$
if $d_{min}(t) \geq 1.3$ **then**
    /* robot moves normally */
    setSpeedFraction(ρ_{normal})
    setSpeedFractionRate($\Delta\rho_{normal}$)
    robot action $\psi \leftarrow \psi_{normal}$
else
    /* stop the robot */
    setSpeedFraction(ρ_{stop})
    setSpeedFractionRate($\Delta\rho_{stop}$)
    robot action $\psi \leftarrow \psi_{stop}$
end

D. ToF Distance Safety Index-based Dynamic SSM

The distance-based dynamic SSM approach using ToF sensor array is similar to the approach used for 2D LiDAR-based static SSM presented in Section 4.1.2.3 B. and is shown in Algorithm 3. This approach also defines the protective/critical safety distance threshold ($d_{j\,stop}$) and the reduced safety distance threshold ($d_{j\,reduce}$) for each ToF sensor array S_j on the j-th robot link to determine the robot action/operation mode ψ. Unlike the 2D LiDAR-based static Tri-SSM safety approach, this is a dynamic Tri-SSM safety setup as the detection region of the ToF sensor arrays changes with the robot pose. The overall minimum distance $d_{min}(t)$ as defined in the previous section with Eq.(4.25a) is used as the input in Algorithm 3. This approach does not take into consideration the velocities of the robot links and human agent/object in the shared workspace. The approach that considers the directed speeds of the robot links is described in the following section.

Algorithm 3: Distance-based Tri-SSM Algorithm using ToF sensor Arrays

Data: safety distance thresholds ($d_{j\,stop}$, $d_{j\,reduce}$), $\forall\ j = \{base, elbow, tool\}$
speed fraction ($\rho_{normal} = 1$, $\rho_{reduce} = 0.5$, $\rho_{stop} = 0$)
speed fraction rates ($\Delta\rho_{normal}$, $\Delta\rho_{reduce}$, $\Delta\rho_{stop}$)
Input : the minimum/shortest distance $d_{j\,min}(t),\ \forall\ j = \{base, elbow, tool\}$
Output: robot action ψ

$d_{min}(t) \leftarrow \min_{j} \{d_{j\,min}\}$
$j^* \leftarrow \operatorname{argmin}_{j} \{d_{j\,min}\}$
if $d_{min}(t) \geq 1.3$ **then**
    /* do nothing */
    robot action $\psi \leftarrow \psi_{normal}$
else
    if $d_{min}(t) \leq d_{j^*\,stop}$ **then**
        /* stop the robot */
        `setSpeedFraction`(ρ_{stop})
        `setSpeedFractionRate`($\Delta\rho_{stop}$)
        robot action $\psi \leftarrow \psi_{stop}$
    else if $d_{j^*\,stop} < d_{min}(t) \leq d_{j^*\,reduce}$ **then**
        /* reduce the robot speed */
        `setSpeedFraction`(ρ_{reduce})
        `setSpeedFractionRate`($\Delta\rho_{reduce}$)
        robot action $\psi \leftarrow \psi_{reduce}$
    else
        /* robot moves normally */
        `setSpeedFraction`(ρ_{normal})
        `setSpeedFractionRate`($\Delta\rho_{normal}$)
        robot action $\psi \leftarrow \psi_{normal}$
    end
end

E. ToF Overall Safety Index-based Dynamic SSM

This approach uses the overall safety index (SI) calculated using the ToF sensor arrays as discussed previously in Section 4.1.2. This safety approach to determine the robot action/operation mode is shown in Algorithm 4. The SI ($si(t)$) for the robot at a given time (t) and normalized DSI and MSI ($\widetilde{dsi_j(t)}, \widetilde{msi_j(t)}$) for all the robot links are calculated using the Eq.(4.7)-(4.12) and Eq.(4.20). The maximum value of $si(t)$ representing the worst-case scenario is $si(t) \leq 2$, as it is a linear combination of the normalized DSI and MSI ($\widetilde{dsi_j(t)} \leq 1$, $\widetilde{msi_j(t)} \leq 1$). Thus, a SI threshold $\mathcal{T}_{SI} \leq 1$ is chosen to incorporate the maximum danger represented by DSI and MSI. The thresholds for DSI and MSI are also defined as $\mathcal{T}_{DSI} \leq \mathcal{T}_{SI}$ and $\mathcal{T}_{MSI} \leq \mathcal{T}_{SI}$, respectively. These thresholds can be interpreted as the maximum allowable component of the overall safety index (si) to determine if a robot action (ψ_{stop}) to stop the robot should be taken. The DSI component of the SI represents an imminent danger of collision in comparison to MSI. Hence, the resulting action as a result of danger indicated by the DSI component is more significant in comparison to MSI. The condition for determining the robot action based on DSI (shown on *Lines[19-22]* of Algorithm 4) is checked after the robot action is determined using MSI (shown on *Lines[10-17]*). An exponential smoothening filter (a first-order low pass filter) is used to reduce changes in the safety index caused due to loss of detected points due to blind-spots, out-of-range observations, and the uncertainty in the calculation of direct speeds v_{o_j} of the detected point on the agent/object. The filter for the SI, $si(t)$ can be defined as

$$si(t)_{filt} = (\alpha)si(t) + (1 - \alpha)si(t - \Delta t) \tag{4.26}$$

where α is the filter coefficient. In other words, $si(t)_{filt}$ is the linear combination of the current safety index $si(t)$ and the previous measured safety index $si(t - \Delta t)$, weighted with the coefficients α and $(1 - \alpha)$. In the case when the overall safety of the robot as defined by $si(t)$ decreases (or danger increases), the current safety index is more critical as the robot action to decrease the overall speed needs to be taken promptly. Hence in this scenario, the filter coefficient is chosen as ($\alpha_D = 0.8$). On the other hand, when the overall safety increases (the danger decreases), the robot can cautiously increase the speed. Hence, the filter coefficient of ($\alpha_I = 0.3$) is chosen.

Algorithm 4: Overall Safety Index-based Dynamic Tri-SSM Algorithm using ToF Sensor Arrays

Data: overall safety index threshold ($\mathcal{T}_{SI}$),
distance and momentum safety index thresholds ($\mathcal{T}_{DSI}$, $\mathcal{T}_{MSI}$),
exponential filter weights ($\alpha_D = 0.8$, $\alpha_I = 0.3$),
speed fraction ($\rho_{normal} = 1$, $\rho_{reduce} = 0.5$, $\rho_{stop} = 0$),
speed fraction rates ($\Delta\rho_{normal}$, $\Delta\rho_{reduce}$, $\Delta\rho_{stop}$)
Input : previous overall safety index at time $(t - \Delta t)$, $si(t - \Delta t)$,
normalized DSI and MSI $(\widetilde{dsi_j(t)}, \widetilde{msi_j(t)})$ $\forall j = \{base, elbow, tool\}$
Output: robot action ψ

/* Overall Safety Index */
$si(t) \leftarrow \max_j \{\widetilde{dsi}_j(t) + \widetilde{msi}_j(t)\}$
$j^* \leftarrow \operatorname{argmax}_j \{\widetilde{dsi}_j(t) + \widetilde{msi}_j(t)\}$
/* exponential filter for smoothening SI */
if $si(t) > si(t - \Delta t)$ **then**
    $si(t)_{filt} = (\alpha_D)si(t) + (1 - \alpha_D)si(t - \Delta t)$
else
    $si(t)_{filt} = (\alpha_I)si(t) + (1 - \alpha_I)si(t - \Delta t)$
end
$si(t - \Delta t) = si(t)_{filt}$
/* robot action */
if $si(t)_{filt} > \mathcal{T}_{SI}$ **then**
    if $\widetilde{msi}_{j^*} > \mathcal{T}_{MSI}$ **then**
        /* stop the robot */
        setSpeedFraction(ρ_{stop})
        setSpeedFractionRate($\Delta\rho_{stop}$)
        robot action $\psi \leftarrow \psi_{stop}$
    else
        /* reduce the robot speed */
        setSpeedFraction(ρ_{reduce})
        setSpeedFractionRate($\Delta\rho_{reduce}$)
        robot action $\psi \leftarrow \psi_{reduce}$
    end
    if $\widetilde{dsi}_{j^*} > \mathcal{T}_{DSI}$ **then**
        /* stop the robot */
        setSpeedFraction(ρ_{stop})
        setSpeedFractionRate($\Delta\rho_{stop}$)
        robot action $\psi \leftarrow \psi_{stop}$
    end

```
24
        /* robot moves normally                                    */
25      setSpeedFraction(ρ_normal) ;
26      setSpeedFractionRate(Δρ_normal) ;
27      robot action ψ ← ψ_normal ;
```

E.1 A Less Conservative version of the Overall Safety Index Algorithm 4 is also used where the robot does not stop but reduces the speed when the danger quantified by MSI (msi_{j^*}) for the j^*-th robot link is greater than the threshold ($\mathcal{T}_{MSI}$). The *Lines [10-17]* of Algorithm 4 can be modified as

```
10    if msi~_{j*} > T_MSI then
          /* reduce the robot speed                                 */
11        setSpeedFraction(ρ_reduce) ;
12        setSpeedFractionRate(Δρ_reduce) ;
13        robot action ψ ← ψ_reduce;
14    else
          /* robot moves normally                                   */
15        setSpeedFraction(ρ_normal) ;
16        setSpeedFractionRate(Δρ_normal) ;
17        robot action ψ ← ψ_normal ;
```

All the SSM safety approaches described in Algorithms 1-4 are defined to determine the robot action/operation mode for an instance of the simulation time (t) and simulation time step(Δt). These safety approaches are used during a standard pick and place experiment and are evaluated based on the safety, performance, and productivity criteria. A representation of Algorithms used for the evaluation and validation is shown in Figure 4.9. The experiment setup and the evaluation criteria are described in the following section.

4.1.3 Experiment Setup and Evaluation Criteria

The V-REP 3D simulation environment (also called CopelliaSim) is used to represent the digital-twin of the HRC setup for the experiment [29, 76]. In the V-REP setup, a standard pick place material handling task of placing ten products in a box is simulated, as shown in

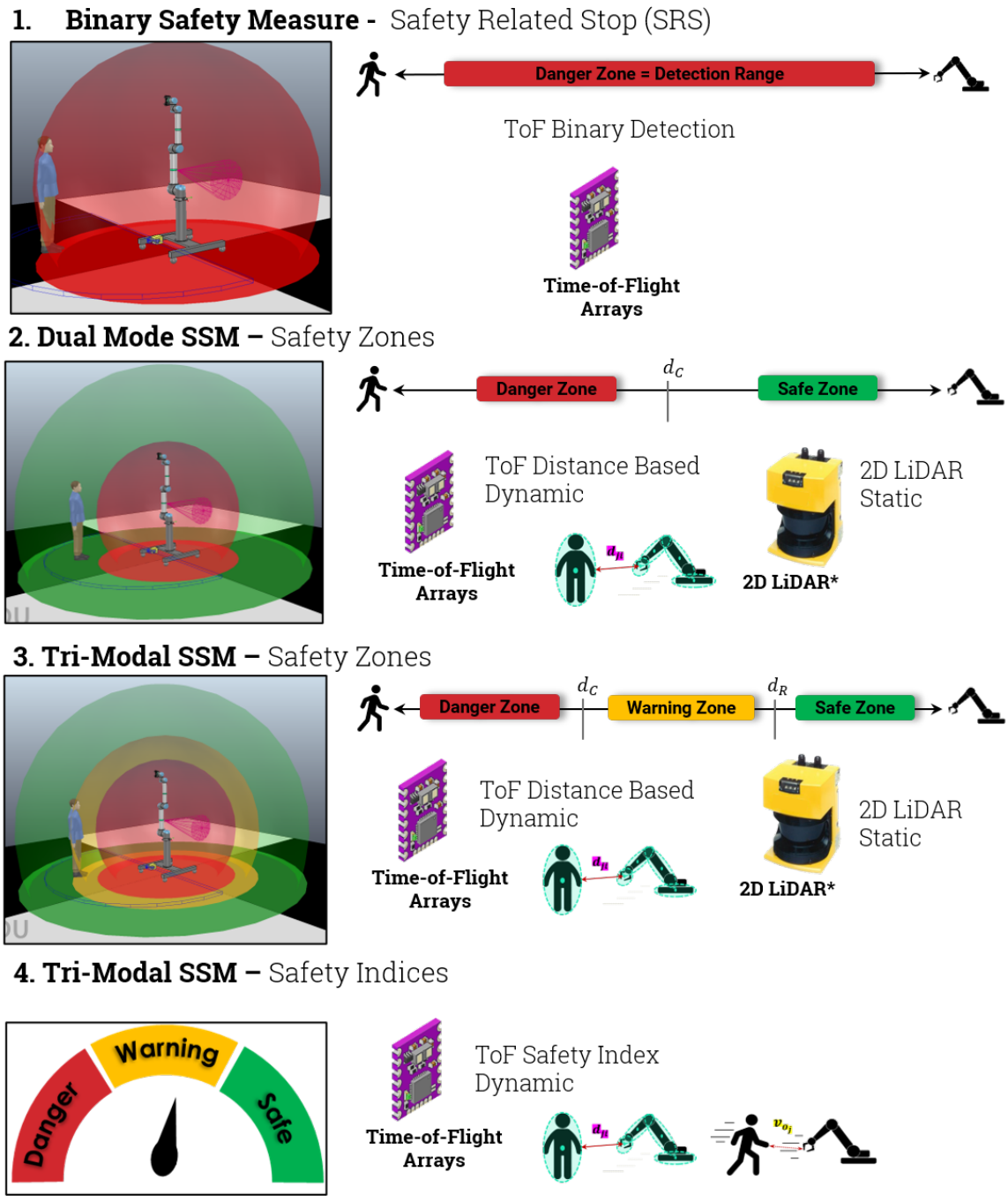


Figure 4.9: A representation of Algorithms used for a comparison of static SSM safety approach implemented using 2D LiDAR and the dynamic SSM approaches using on-robot ToF sensor arrays.

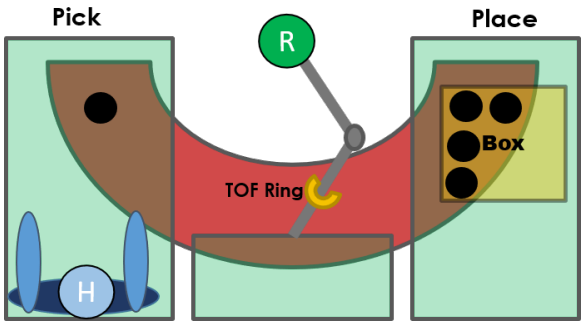
Figure 4.10. The robot movement involves moving the base joint 180° degrees between the pick and place positions on the tables. This task was chosen as the base joint of a robot has the largest braking distance when moving at high joint speeds [59] and can be considered a good test case for validating SSM safety approaches. This results in a radial motion of the Tool-Control-Point(TCP) or the end-effector as it moves between the pick and place locations during the task.

This setup is a co-existence form of human-robot collaboration where the human and robot share a workspace but do not interact directly. The objective of the robot behavior is the completion of the task while ensuring human safety using the SSM safety approaches discussed here. In order to validate and compare the SSM safety approaches for proximity-based sensors, it requires the use of controlled physical avatars that can move in repeatable trajectories to maintain the same human interaction as suggested in [46, 85]. In this setup, a simulated human-avatar follows a '∞' (an overlapped infinity or figure eight) shaped path overlapping the robot's operating workspace, as shown in Figure 4.10(b) [6]. The human path is fixed to simulate similar human interference of the robot task.

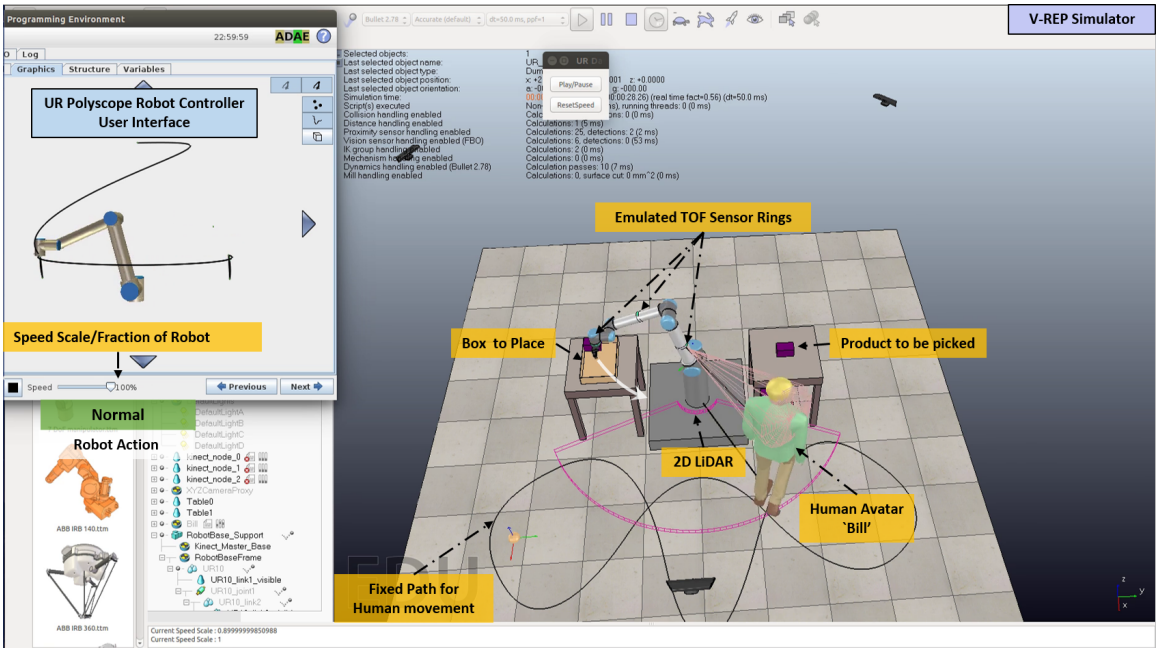
The simulated UR10 robot in the digital-twin is connected to the UR10 robot controller for robot action control to ensure the robot response time and maintain the integrity of the task. The robot is made to *hallucinate* the presence of the human in the real world by moving the human avatar in digital-twin on the fixed path. This ensures the human interaction across all safety approaches is the same. For communication, the Robot Operating System (ROS) was used to update the robot position in the simulated environment as reported by the UR Polyscope software. The V-REP simulation mimicked the execution of the task on the robot controller.

The ToF sensor array on the UR10 robot links are simulated using ray-based proximity sensor models emulated in the physics-engine 3D environment of V-REP [29, 76]. Each sensor node is represented with a cone-shaped proximity sensor with detection range (height of the cone) of $0.03m$ to $1.3m$ and a FOV angle of 25° degrees (refer Figure 4.2 and Figure 4.8). The detection rate of the simulated sensor is based on the V-REP simulation time-step ($\Delta t = 50ms$), which is 50ms or an update rate of 20Hz.

The 2D scanning LiDAR is also simulated using the emulated sensor model for a SICK 2D LiDAR scanner provided in the V-REP (CopelliaSim) simulation software. The 2D scanning LiDAR is placed at the base of the robot closer to the floor. The detection zone



(a)



(b)

Figure 4.10: (a) A schematic of the standard robot pick and place task simulated for the experiment. (b) The experiment setup represented in the simulation where a human-avatar moves on a fixed path that overlaps with the robot motion while performing the pick and place task.

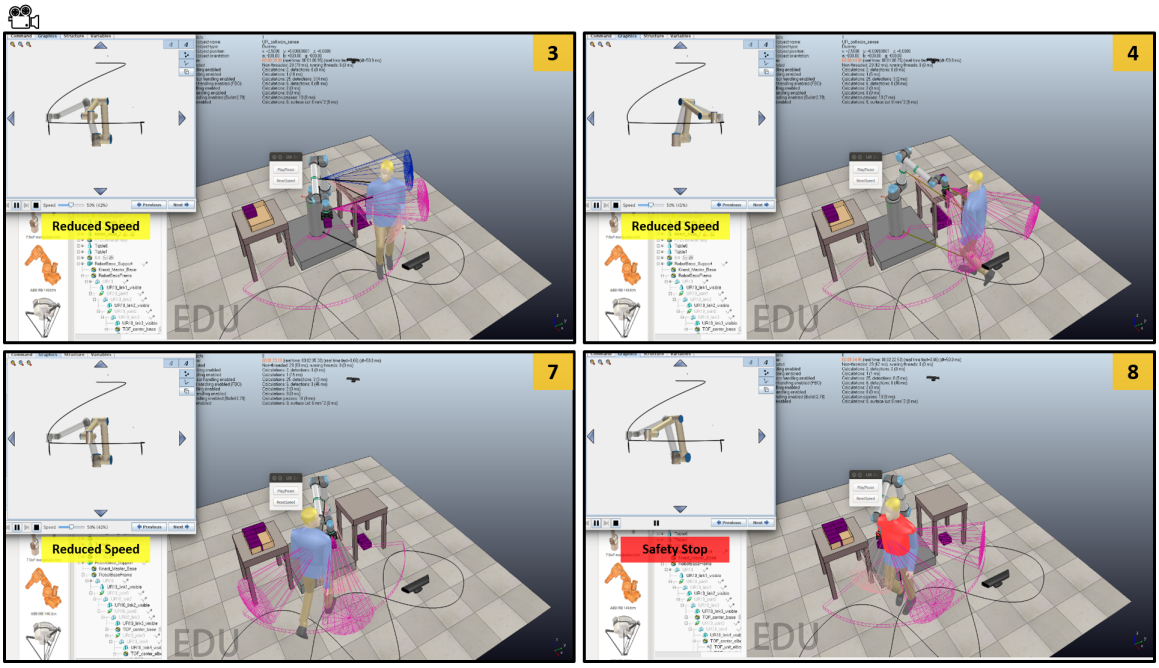


Figure 4.11: Frames of a video for a ToF experiment during the iteration 3,4,7 and 8. The iteration 8 shows Human-Robot collision. Video Link

angle and range of the 2D LiDAR are set to 180° degrees and 3m, respectively in order to cover the robot operating workspace and the collaborative workspace of the experiment setup (refer Figure 4.10 and Figure 4.7).

The behavior of the robot for different safety approaches with the minimum distance and directed speeds calculated based on the human avatar is analyzed by running experiments with varying parameters of speed fraction, speed fraction rates, and thresholds of distance, velocity and safety indices. Frames of an experiment running are shown in Figure 4.11.

The evaluation and validation criteria of the experiments is detailed in the following section.

4.1.3.1 Evaluation & Validation Criteria

The SSM safety approaches for 2D LiDAR and simulated on-robot ToF sensor arrays are evaluated based on the criteria of *safety*, *performance*, and *production* during a single robot

task. The evaluation metric has been presented in Table 4.1. The objective of this evaluation and validation is a comparison of static SSM safety approach implemented using 2D LiDAR and the dynamic SSM approaches using on-robot ToF sensor arrays.

As this is a co-existence HRC setup, the baseline for comparison of the safety approaches is when there is no human interaction with the robot, and the human/agent is always out of detection range (shown in Figure 4.12). The different safety approaches as described in Section 4.1.2.3 that are compared are:

- **No Human-Robot Interaction - *NoHRI***: The robot performs the task without a human in the workspace. The results of these are taken as baseline of comparison for all the rest of the experiments. The video showing this setup can be viewed at https://youtu.be/uqOguEDdlXM.
- **Static Awareness (Static SSM)- 2D LiDAR - *Static***: The robot performs the task with the conventional static SSM safety approaches used in industry based on the distance between human and robot as measured by the 2D LiDAR. The experiments with dual-mode SSM and Tri-SSM are implemented as described in Section 4.1.2.3 B. . The video showing this setup can be viewed at https://youtu.be/Bhklwi8JMek.
- **Dynamic Awareness (Dynamic SSM) - *ToF_***: the robot performs the task with the dynamic SSM safety approaches implemented using ToF sensor arrays placed on the robot as described in Section 4.1.2.3. The different ToF-based approaches compared are
  - **Binary** : If a human agent within the detection range of any of the ToF sensor arrays, the robot stops. This is a very conservative dual-mode dynamic SSM safety approach (described in Section 4.1.2.3C.). The video showing this setup can be viewed at https://youtu.be/o73g82O9zcs.
  - **Distance-Based** : A Tri-SSM safety approach based on the minimum distance measurements from all the ToF sensor arrays (described in Section 4.1.2.3D.). The video showing this setup can be viewed at https://youtu.be/WKg5gIANYy8.
  - **Overall Safety Index**: A Tri-SSM safety approach based on the overall safety index dependent on the human-robot minimum distance and directed speeds (described in Section 4.1.2.3E.). The video showing this setup can be viewed at https://youtu.be/JukGg2uKiIY.

The significance of the evaluation and validation criteria are explained as follows:

- **Robot Idle Time** is the total time when the robot speed is zero. This time is the total time when the robot is not moving because it is unable to move due to the action to stop

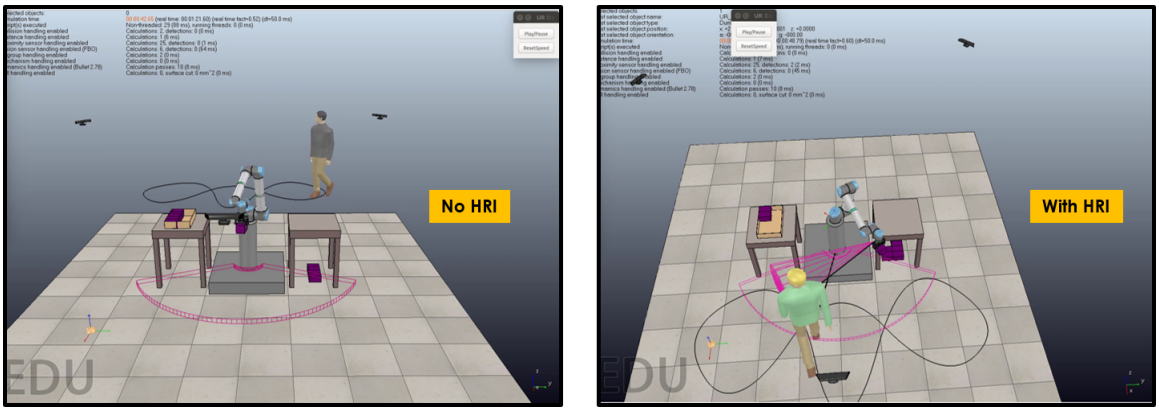


Figure 4.12: The *NoHRI* experiment setup where there is no human interaction with the robot and the human/agent is always out of detection range.

Table 4.1: A evaluation criteria metric for the proposed system.

CRITERIA	ROBOT OPERATION SCENARIO		
	No HRI (Base Line)	**Static SSM (2D LiDAR)**	**Dynamic SSM (ToF Sensor Arrays)**
Safety	•Safety Stop and Reduce Speed Events •Number of conflicts b/w human and robot. •End-effector/TCP Velocity before and at collision.		
Performance	•Sum of the derivatives of the robot joint torques or Total Sum Jerk. •Robot Idle Time		
Productivity	•Time taken to complete the task.		

the robot ψ_{stop} as determined by the safety approach. This reduces the productivity of the task.

- **Number of Collisions** between the human and the robot. A higher number of collisions would result in less trust and comfort for the human. The velocity before and at the time of collision indicates the severity of the collision. It tells whether the robot was reactive enough to stop or reduce its speed in time.
  (*Note: Collision is defined here as the event where the robot velocity is not zero when the minimum distance between human and robot is less than the minimum acceptable/threshold distance.*)
- **Safety Stop Events** where the robot is instructed to stop. The number of instances when the robot action ψ_{stop} was taken.

- **Reduced Speed Events** where the robot is instructed to perform the task at reduced speeds. The number of instances when the robot action $\psi_{reduced}$ was taken.
- **Sum Jerk** indicates the smoothness between transitions of speed change. The lower sum of jerks and smoother robot motion helps increase human comfort and trust in automation. Moreover, less jerks cause less wear and tear of the robot. Here, this is calculated as the sum of the derivatives of the reported robot joint torques. The derivative of the joint torque is called Rotatum and not 'Jerk'. But for easy understanding of the significance of this criteria, it is termed as 'Sum Jerk'.
- **Productivity** is determined based on the time taken to complete task, which is performing pick and place task ten times for placing the ten objects in a box. The time taken is compared to the time taken to complete the same task with no human interaction (*NoHRI* scenario).

Similar criteria have been used to evaluate human-robot collaboration systems in the works like [65, 66, 67].

4.1.3.2 Assumptions

The ToF sensor ring, along with the detection of human, would detect the robot, objects, and the restricted workspace containing the tables, robot base, and floor. For detection of human-agent only, the readings of sensor nodes detecting other entities in the workspace are masked. In other words, the human-agent is the only detectable entity in the digital-twin. This is achieved in the simulation as V-REP uses a physics engine to determine the intersections between meshes and the cone detection area of ToF sensor nodes [29, 76]. In order to keep the human interaction the same over the experiments, the human is moved at $0.576m/s$ on a fixed predefined path. The total time, which is the sum of the processing time, the controller reaction time, and the robot stopping time for a robot to execute a safety stop (the robot action ψ_{stop}) is assumed to be $\approx 125ms$. For a max TCP velocity of $1.72m/s$ during the task, this results into a braking/stopping distance of $\approx 0.3m = (1.72m/s + 0.576m/s) * 0.125s$. This is the distance traveled by the end-effector from the time the robot stop action is triggered to the robot actually coming to a halt.

4.1.4 Results and Observations

The tabulated results in Table 4.3 are for the experiments with parameters shown in Table 4.2. Table 4.2 lists the speed fraction parameters (ρ, $\Delta\rho$) and the thresholds used in multiple experiments with different safety approaches discussed previously in Section 4.1.2.3. The thresholds for the 2D LiDAR-based static SSM ($d_{2D_{reduce}}$, $d_{2D_{stop}}$) for three experiments performed are listed. The first two experiments using the 2D LiDAR implement a dual mode SSM where the robot action ($\psi_{reduced}$) is not considered. For ToF distance-based safety approach, the distance safety thresholds ($d_{j\,reduce}, d_{j\,stop}$) for each ToF sensor array of the j-th robot link to implement SSM are listed in Table 4.2. The thresholds for the ToF overall safety index-based safety approach shown in Table 4.2 for SI, DSI and MSI are $\mathcal{T}_{SI}, \mathcal{T}_{DSI}$ and $\mathcal{T}_{MSI}$, respectively. The distance and velocity thresholds ($d_{min_{thresh}}$, $v_{max_{thresh}}$) that are used to normalize the DSI and MSI according to the Eq.(4.20).

The highlighted rows of the Table 4.3 shows the best result for each safety approach. In comparison to the baseline *NoHRI*, the best *2D LiDAR-based Static* Tri-SSM safety approach (experiment 3) increases the task completion time by $\approx$ 39%, the total sum jerk for robot base joint $J0$ by $\approx$ 66% and the robot idle time by $25s$. The *ToF Binary Detection* approach is too conservative as it increases the detection range of the robot beyond its operating workspace and the task completion time is increased $\approx$ 184%, almost three times. The best *ToF Distance* based dynamic Tri-SSM safety approach (experiment 4) results increase the task completion time by $\approx$ 12%, the total sum jerk for robot base joint $J0$ by $\approx$ 23.5% and the robot idle time by $\approx$ $1s$. This is a significant improvement over the *2D LiDAR-based Static* Tri-SSM safety approach. This is the fastest completion time for a task with no collisions with the human. The observed overall movement was smoother, as the number of stops were less. However, this safety approach required tuning of the thresholds and the speed fraction parameters to determine an optimal SSM safety setup. This safety approach is more prone to collisions, as seen with other experiments due to sensor blind-spots and reaction time of the robot.

The best *ToF Safety Index* based dynamic Tri-SSM safety approach (experiment 2) results increase the task completion time by $\approx$ 16%, the total sum jerk for robot base joint $J0$ by only $\approx$ 1%, and the robot idle time by $\approx$ $8s$. This approach is better in terms of the conventional *2D LiDAR-based Static* SSM approach in all regards. In comparison

Table 4.2: List of parameters for the experiments to evaluate 2D LiDAR-based Static SSM and ToF sensor-based Dynamic SSM safety approaches.

Experiment Name	No	Speed Fraction Parameters						Thresholds								
		Normal		Safe Stop		Reduce		Distance(reduce,stop) (m)			Velocity (m/s)			Safety Indexes		
		ρ	$\Delta\rho$	ρ	$\Delta\rho$	ρ	$\Delta\rho$	Base	Elbow	Tool	Base	Elbow	Tool	DSI	MSI	SI
NoHRI	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2D LiDAR Static	1	1	**0.2**	0	**0.5**	-	-	**(reduce: - , stop: 1.2)**			-	-	-	-	-	-
	2	1	0.2	0	**0.3**	-	-	(reduce: - , stop: 1.2)			-	-	-	-	-	-
	3	1	0.2	0	**0.5**	**0.5**	**0.5**	**(reduce: 1.2, stop: 0.9)**			-	-	-	-	-	-
ToF Binary Detection	1	1	0.2	0	0.5	-	-	-	-	-	-	-	-	-	-	-
ToF Distance-based	1	1	0.2	0	0.5	-	-	(- ,0.6)	(- ,0.6)	(-,0.6)	-	-	-	-	-	-
	2	1	0.2	0	0.5	-	-	**(- ,0.3)**	**(- ,0.3)**	**(- ,0.3)**	-	-	-	-	-	-
	3	1	0.2	0	0.5	-	-	(- ,0.3)	**(- ,0.45)**	**(- ,0.6)**	-	-	-	-	-	-
	4	1	0.2	0	0.5	**0.5**	**0.2**	**(0.6,0.3)**	**(0.6,0.3)**	**(0.6,0.3)**	-	-	-	-	-	-
	5	1	0.2	0	**0.3**	0.5	0.2	(0.6,0.3)	(0.6,0.3)	(0.6,0.3)	-	-	-	-	-	-
	6	1	0.2	0	0.3	0.5	0.2	**(0.75,0.3)**	**(0.75,0.45)**	**(0.75,0.6)**	-	-	-	-	-	-
ToF Safety Index	1	1	0.2	0	0.5	0.5	0.2	**0.3**	**0.3**	**0.3**	**0.3**	**0.5**	**0.5**	**0.8**	**0.9**	**1**
	2	1	0.1	0	**0.2**	0.5	**0.1**	0.3	0.3	0.3	0.3	0.5	0.5	0.8	0.9	1
	3	1	0.1	0	0.2	0.5	0.1	0.3	0.3	0.3	0.3	0.5	0.5	**0.9**	0.9	1
	4	1	0.1	0	0.2	0.5	0.1	0.3	0.3	0.3	0.3	0.5	0.5	0.8	**1**	1
	5*	1	**0.2**	0	**0.5**	0.5	**0.2**	0.3	0.3	0.3	0.3	0.5	0.5	0.8	**0.7**	1
	6*	1	**0.1**	0	**0.2**	0.5	**0.1**	0.3	0.3	0.3	0.3	0.5	0.5	**0.7**	**0.7**	1

Table 4.3: Results for the experiments to evaluate 2D LiDAR-based Static SSM and ToF sensor-based Dynamic SSM safety approaches.

Name of Experiment	No	Sum Jerk (Joint Torque Derivative)			Robot Idle Time (sec)	# Stop Event	# Reduced Speed Event	Task Completion Time (sec)	Collision		
		J0	J1	J2					# Collision	Velocity before Collision(m/s)	Velocity at Collision (m/s)
NoHRI	**1**	4986.98	2785.43	1500.26	40.195	0	0	125.053	0	0	0
2D LiDAR Static	1	8508.74	3550.74	1866.11	131.272	1363	0	227.732	0	0	0
	2	**4987.08**	**2963.41**	**1604.27**	134.791	1363	0	228.067	0	0	0
	3	8273.05	3512.16	1983.47	**65.178**	**331**	697	**173.601**	0	0	0
ToF Binary Detection	1	18234.38	4594.81	2367.00	254.803	3068	0	355.078	0	0	0
ToF Distance-based	1	8312.79	3374.64	1773.78	60.2	293	0	149.285	1	**0.875**	**0.0946**
	2	6077.06	2934.62	1583.2	**38.026**	**53**	0	**130.135**	2	**1.72**	**1.72**
	3	8368.06	3292.93	1790.22	60.751	358	0	155.4	1	**1.58**	**0.77**
	4	6162.24	3066.85	1683.37	41.042	80	245	140.107	0	0	0
	5	5178.35	3063.19	1721.66	47.662	100	210	141.328	1	**1.28**	**0.507**
	6	**4853.42**	**3055.15**	**1699.97**	51.086	226	**202**	149.03	0	0	0
ToF Safety Index	1	6461.57	3298.90	1830.18	54.68	**162**	124	149.47	0	0	0
	2	5036.05	3091.80	1703.90	**48.60**	171	115	**145.18**	0	0	0
	3	**4953.57**	**3086.49**	**1695.82**	54.187	166	131	146.93	1	**1.12**	**0.72**
	4	5075.53	3075.51	1694.29	53.96	217	**114**	148.90	1	**1.06**	**0.56**
	5*	8231.89	3477.66	1882.22	55.69	217	138	149.005	0	0	0
	6*	5146.08	3135.23	1758.00	52.097	194	115	148.937	0	0	0

to the *ToF Distance*, the robot moves lot more smoothly. Although the time taken is 4s more, it compensates with a smoother (less jerky) motion and results in less number of collisions. The experiments, 5* and 6*, are the results of robot actions taken as per the less conservative Algorithm 4 (refer Section 4.1.2 E.), but gives similarly good results.

A chart representing the summarized results of these safety approaches is shown in Figure 4.13.

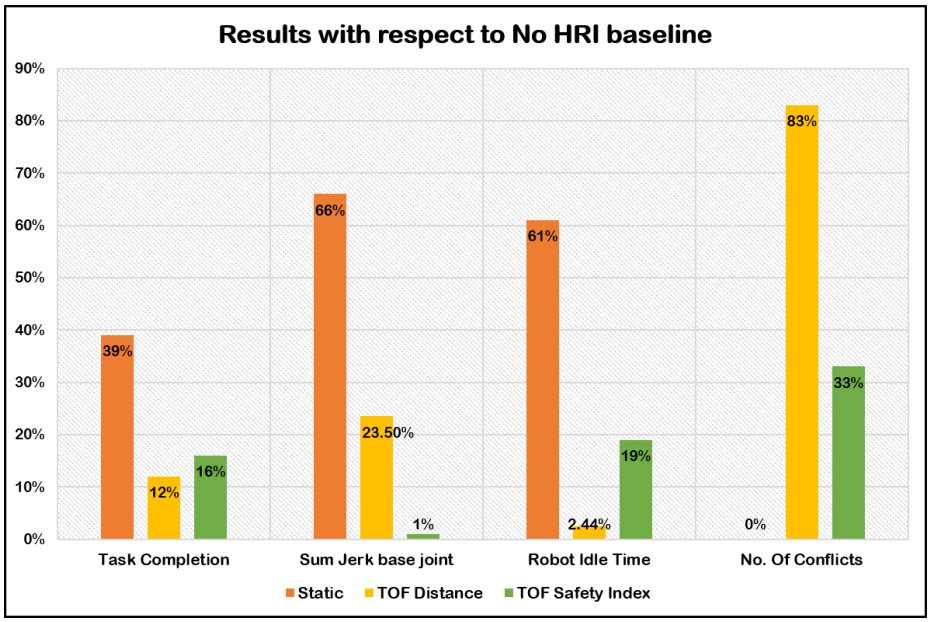


Figure 4.13: A chart summarizing the comparison of the evaluation of the 2D LiDAR static SSM, ToF distance-based dynamic SSM and ToF safety index-based dynamic SSM safety approaches.

4.1.5 Limitations

The ToF sensor ring in simulation is programmed to detect only the human avatar and ignore other objects and robot-self in the workspace. However, that would not be the case in the physical world. The sensors would detect a human, objects, robot links, floor, and other objects on the robot itself and restricted workspace, resulting in false robot stops and motion. The transition between the different robot actions/operation modes is done by assuming a linear change of the speed fraction (ρ) using ($\Delta\rho$). Sudden changes in state would result in higher jerks and protective stops due to joint torque spikes. Hence, there is a need for an online-acceleration profile for the implementation of a non-linear change of the overall speed for smoother transitions between robot speeds. The results shown are for similar human interaction moving at a consistent speed of $0.576 m/s$, which is not the case in an actual industrial setup.

The parameters of speed fraction rate and the thresholds for DSI and MSI are less structured and arbitrarily chosen as there is not a clear defined way to calculate them. This setup did not consider the calculations of directed speeds in cases of loss of detection due to

blind-spots or sudden detection when a human-agent is withing the sensing zones of the ToF sensor arrays. As a result, the uncertainty and sudden changes in SI were mitigated by using an exponential filter. The rate of detection of the simulated ToF sensor nodes is limited by the simulation time step and the number of proximity sensor distance calculations performed during each simulation time step. As the number of sensing components in simulation increases, the overall computation required also increases and the update rate of the distance measurements suffers. The actual hardware of the ToF sensor node has an update rate of 27Hz, higher than V-REP and with minimal computational overhead.

4.1.6 Intermediate Conclusion

In this first part of the chapter, the use of Time-of-Flight sensor arrays as on-robot ranging sensors mounted on the centers of robot links for implementation of a dynamic SSM safety configuration is validated using a digital-twin. The methodology of ellipsoidal approximation for a 6DOF robot as a 3-DOF for placement of the ToF sensor arrays is successfully validated. The relation of minimum distance ($d_{j\,min}$) and directed speeds (v_{o_j}) for the robot links using the ToF sensor distance measurements and the robot joint pose and velocities($\boldsymbol{q}, \dot{\boldsymbol{q}}$) is formulated. These are used to define safety indices for implementing dynamic SSM.

Under the assumption of consistent human interactions, the simulation results suggest a significant benefit of dynamic SSM using on-robot ToF sensor arrays in terms of safety, performance, and productivity during the HRC task in comparison to statically aware SSM using a 2D LiDAR. Using human-robot separation distance thresholds for the dynamic Tri-SSM resulted in the lowest productivity loss but was shown to be more prone to conflicts. The overall safety index (SI) based safety approach that leverages both human-robot separation distance and human-robot directed speed resulted in a more consistent and smoother robot movement with minimum conflicts. The results of these experiments validate that using on-robot ToF-based sensor arrays for SSM safety setup is a viable alternative or an addition to current the 2D LiDAR-based safety setups for optimizing productivity while ensuring human safety during tasks with frequent human interaction.

The next step was the design and deployment of a prototype for the ToF sensor array

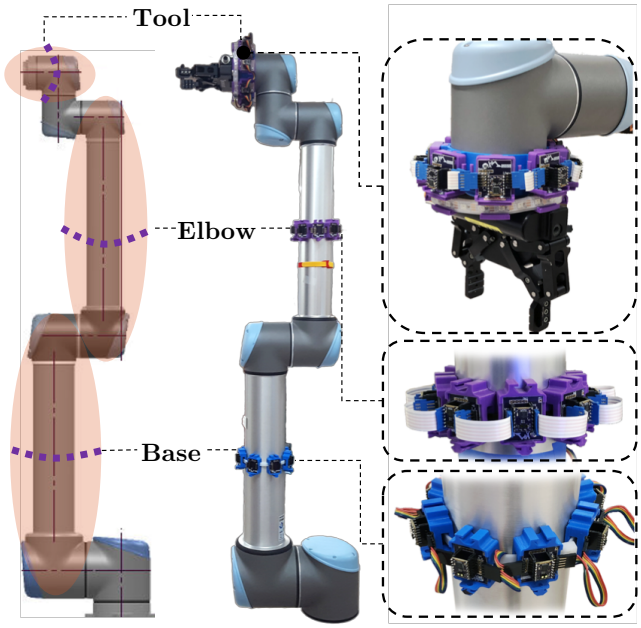


Figure 4.14: The ellipsoidal approximation for the 6-DOF UR10 robot links (base, elbow and tool), where the 8-node ToF sensor array ring prototypes are placed [1].

rings on the UR10 robot links. A dynamic SSM safety configuration based on the industrial SSM formulation was implemented. The details of the hardware and software for the prototype of an 8-node ToF sensor array used in this research are detailed in Chapter 6. The implementation of a dynamic SSM safety setup using the 8-node ToF sensor array prototype is detailed in the next part of this chapter.

4.2 Implementation of Dynamic SSM using ToF Sensor Arrays

In this section, a dynamic SSM safety approach using on-robot ToF sensor arrays is implemented for use in industrial HRC. A prototype of the 8-node ToF sensor array is developed and placed on the UR10 robot link, as shown in Figure 4.14 (discussed in detail in Chapter 6).

An SSM formulation based on the standards [12,43] and research [13,46,85] for defining the thresholds for dynamic SSM implementation using ToF sensor arrays is presented.

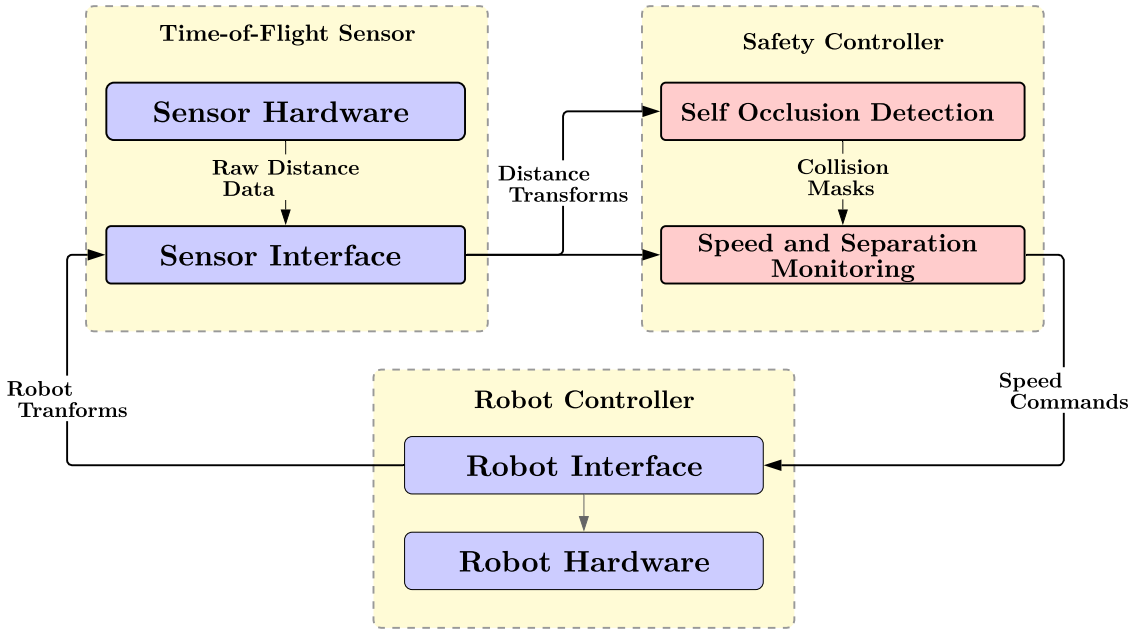


Figure 4.15: A High-Level Block Diagram representing the dynamic SSM setup using ToF sensor arrays. It must be noted that the sensor interface is also responsible for merging the robot kinematic chain with the raw distances provided by the sensors.

A methodology of self-occlusion check to disregard distance measurements of restricted workspace and robot-self during SSM is described. A high-level block diagram of the proposed ToF sensor array based dynamic SSM is shown in Figure 4.15. An evaluation in terms of safety, performance, and productivity of different variations of the ToF-based dynamic SSM safety setup and the simulated 2D LiDAR-based setup is reported. The experiment setup used here is similar to the one described previously in Section 4.1.3. Unlike a simulated human-avatar, a human operator interacts physically with the safety setup. For the experiments, a motion-capture system has been integrated with the digital-twin setup to track the human-position in the workspace, use the distance information to simulate an ideal behavior of a ToF sensor array and provide ground-truth for comparison of distance measurement accuracy.

Many SSM approaches have been developed and implemented by others using various sensor modalities such as the 2D LiDAR [85,111], 3D depth camera [44,47,84], IR ranging sensors [19] and others as described in [13, 18]. The approach presented here is the first

novel attempt to use on-robot Time-of-Flight-based sensor arrays to achieve dynamic Tri-Modal SSM for a collaborative robot. The collaborative robot UR10 used to implement and test this safety setup is a 6-DOF robot manipulator similar to an industrial robotic arm with maximum operation reach of $1.3m$ and a payload capacity of $10kg$. It is also one of the most commonly used robots in the industry for HRC applications. The ToF-based dynamic SSM safety setup described here was not implemented to be dependent on a type of robotic manipulator, and can be used for proximity sensing based safety setups for all robots. It is implemented as a replacement or auxiliary of the conventional 2D scanning LiDAR.

The approach presented here is organized as follows: the SSM formulation for the safety distance thresholds, the minimum distance and directed speeds calculation using the ToF sensor arrays, self-occlusion check, the dynamic Tri-SSM algorithm to determine the robot action/operation mode ($\psi = \{\psi_{normal}, \psi_{reduced}, \psi_{stop}\}$) and robot control and interface to execute the robot actions under timing constraints.

In the next section, the SSM formulation to determine distance safety thresholds is presented.

4.2.1 SSM Formulation

This SSM formulation presented here describes the formulation of a critical/protective safety distance (PSD) threshold as a function of the human and robot speeds, robot reaction times, robot stopping time and braking distance. Unlike the overall safety index SI implemented previously in Section 4.1.1.3 that quantifies the danger represented by the DSI and MSI separately, this formulation incorporates it in defining an adaptive PSD threshold. A reduced safety distance (RSD) threshold based on the SSM formulation is also defined for its use in Tri-Modal SSM.

4.2.1.1 Critical/Protective Safety Distance (PSD)

A robot stop (ψ_{stop}) is triggered at time t when the human-robot-separation distance ($d(t)$) is less than or equal to the PSD threshold ($d_C(t_0)$) to avoid a human-robot collision. A time interval diagram representing the SSM formulation for defining the PSD between a human and a robot at time (t_0) is shown in Figure 4.16 [10].

According to [12, 43] ISO standards and [46], the critical/protective safety distance

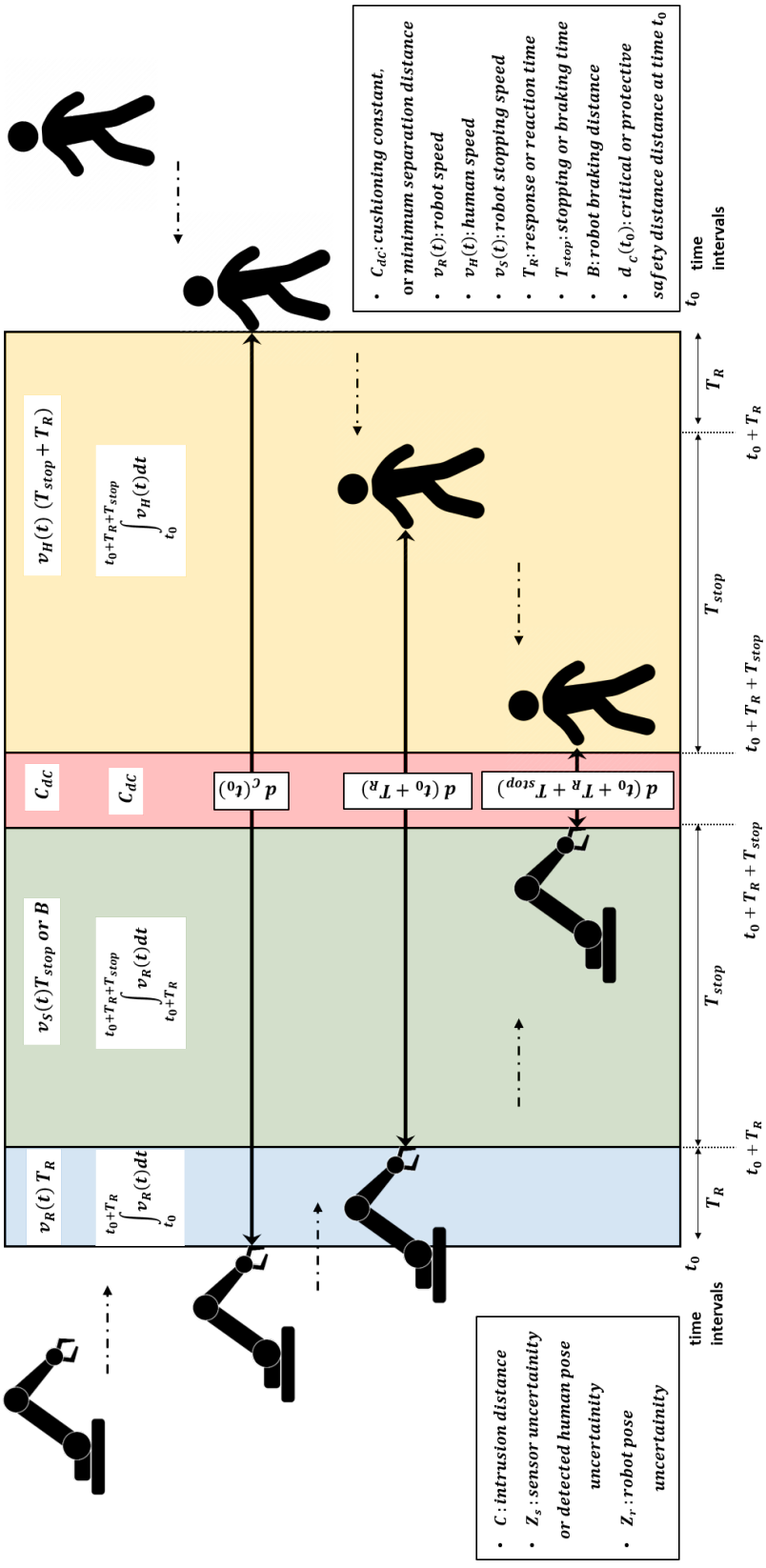


Figure 4.16: A time interval diagram representing the SSM formulation for defining the critical/protective safety distance between a human and a robot at time t_0.

(PSD) $d_C(t_0)$ (also known as minimum protective distance) at a given time t_0 can be defined as

$$\begin{aligned} d_C(t_0) \geq & \underbrace{\left(\int_{t_0}^{t_0+T_R+T_{stop}} v_H(t)\right)}_{\text{human dist.}} + \underbrace{\left(\int_{t_0}^{t_0+T_R} v_R(t)\right)}_{\text{robot reaction dist.}} \\ & \underbrace{+\left(\int_{t_0+T_R}^{to+T_R+T_{stop}} v_s(t)\right)}_{\text{robot stopping dist.}} + \underbrace{(C + Z_s + Z_r)}_{\text{cushioning constant}} \end{aligned} \tag{4.27}$$

where $v_H(t)$ is the speed of the human, and $v_R(t)$ is the robot speed. T_R is the robot's reaction time, a delay in starting to decelerate with a $v_S(t)$ speed profile. T_{stop} is defined as the total stopping time. The terms C, Z_s & Z_r represent the intrusion distance, the operator position uncertainty, and the robot pose uncertainty, respectively. Going forward, these terms will be combined as a constant C_{dC}, *cushioning constant*. The standards and research do not clearly specify whether the human and robot speeds are speeds towards each other or the magnitude of the velocities. It is also unclear whether the separation distance is measured from the base, end-effector/TCP, or between the closest points on the human and robot. The SSM formulation in Eq.(4.27) has been interpreted and used differently according to the task, the sensor used, and the information available. In industry, the linear form of the Eq.(4.27) defined as

$$\begin{aligned} d_C(t_0) \geq & \underbrace{v_H(t)(T_R + T_{stop})}_{\text{human distance}} \quad \underbrace{+ v_R(t)T_R}_{\text{robot reaction distance}} \\ & \underbrace{+ v_S(t)T_{stop}}_{\text{robot braking distance}} \quad \underbrace{+ C_{dC}}_{\text{cushioning constant}} \end{aligned} \tag{4.28}$$

has been commonly used. The human-robot separation distance is calculated from the robot end-effector/TCP, and the robot speed is the magnitude of the velocity of the robot end-effector as it represents the fastest point on the robot in most cases. The PSD threshold is a sum of the distance travelled by the human ($v_H(t)(T_R + T_{stop})$), the robot ($v_R(t)T_R + v_S(t)T_{stop}$)) for the time interval ($T_R + T_{stop}$), and a cushioning constant C_{dC}. The cushioning constant, C_{dC}, is a buffer distance to compensate for uncertainty in sensor

measurements and robot pose. A conservative estimate of $d_C(t_0)$ can be calculated using the maximum values of v_H ($V_{H_{max}}$, the maximum speed of the human/agent or its body part), the velocity of the robot at time (t_0), the braking distance B and the stopping time T_{stop} (defined in the robot specifications for the robot moving at maximum speeds, carrying a maximum load with all robot joints extended, which is the robot end-effector has the highest momentum [59, 85]). The Eq.(4.27) is rewritten as

$$d_C(t_0) \geq \underbrace{V_{H_{max}}(T_R + T_{stop})}_{\text{human distance}} \underbrace{+ v_R(t_0)T_R}_{\text{robot reaction distance}} \underbrace{+ B}_{\text{robot braking distance}} \underbrace{+ C_{dC}}_{\text{cushioning constant}} \tag{4.29}$$

This form depends solely on the magnitude of the robot end-effector/TCP velocity $v_R(t_0)$ at time instance t_0 that is available from the robot controller. If directed human-robot speed (v_o) is considered the SSM formulation would be

$$d_C(t_0) \geq \underbrace{(v_o(t_0)T_R + v_o(t)T_{stop})}_{\text{human distance}} \underbrace{+ v_R(t_0)T_R}_{\text{robot reaction distance}} \underbrace{+ B}_{\text{robot braking distance}} \underbrace{+ C_{dC}}_{\text{cushioning constant}} \tag{4.30}$$

4.2.1.2 Reduced Safety Distance (RSD)

The Eq.(4.27)-(4.30) define PSD for the implementation of a dual-mode SSM (the robot stops (ψ_{stop}) or moves normally (ψ_{normal})). For Tri-Modal SSM, a reduced safety distance (RSD) such that $d_R(t_0) \geq d_C(t_0)$ is defined. A time interval diagram for illustrating the reduced safety distance (RSD) at time t_0 is shown in Figure 4.17. The RSD is the sum of the distance traveled by the robot and human for the time interval ($T_R + T_{red}$) at the end of which the robot slows down to half its speed ($v_R(t_0 + T_R + T_{red}) = \frac{v_R(t_0)}{2}$), and the PSD

($d_C(t_0 + T_R + T_{red})$) at the end of the interval. The RSD formulations (based on Eq.(4.27)-(4.28)) are defined as

$$d_R(t_0) \geq \underbrace{(\int_{t_0}^{t_0+T_R+T_{red}} v_H(t))}_{\text{human dist.}} + \underbrace{(\int_{t_0}^{t_0+T_R} v_R(t))}_{\text{robot reaction dist.}} + \underbrace{(\int_{t_0+T_R}^{to+T_R+T_{red}} v_{red}(t))}_{\text{reducing speed dist.}} + \underbrace{d_C(t_0 + T_R + T_{red})}_{\text{PSD}} \tag{4.31a}$$

$$d_R(t_0) \geq \underbrace{v_H(t)(T_R + T_{red})}_{\text{human distance}} + \underbrace{v_R(t)T_R}_{\text{robot reaction distance}} + \underbrace{v_{red}(t)T_{red}}_{\text{reducing speed dist.}} + \underbrace{d_C(t_0 + T_R + T_{red})}_{\text{PSD}} \tag{4.31b}$$

where the PSD at time $(t_0 + T_R + T_{red})$, $d_C(t_0 + T_R + T_{red})$ is

$$d_C(t_0 + T_R + T_{red}) \geq \underbrace{v_H(t_0 + T_R + T_{red})(T_R + T_{stop})}_{\text{human distance}} + \underbrace{\frac{v_R(t_0)}{2}T_R}_{\text{robot reaction distance}} + \underbrace{B}_{\text{robot braking distance}} + \underbrace{C_{dC}}_{\text{cushioning constant}} \tag{4.32}$$

The stopping time of the robot (T_{stop}) and the braking distance (B) will also be less as the robot speed at the time of triggering stop action ($t_0+T_R+T_{red}$) is reduced to half of $v_R(t_0)$. These robot braking parameters can be calculated by doing robot braking experiments, as described in [46]. However, for a more conservative definition of PSD, the (T_{stop}, B) are not changed.

The RSD based on directed human-robot speeds ($v_o(t)$) can be defined as

$$d_R(t_0) \geq \underbrace{(v_o(t_0)T_R + v_o(t)T_{red})}_{\text{human distance}} + \underbrace{v_R(t_0)T_R}_{\text{robot reaction distance}} + \underbrace{v_{red}(t)T_{red}}_{\text{reducing speed dist.}} + \underbrace{d_C(t_0 + T_R + T_{red})}_{\text{PSD}} \tag{4.33}$$

This equation can be rearranged to represent the total human-robot relative distance changed

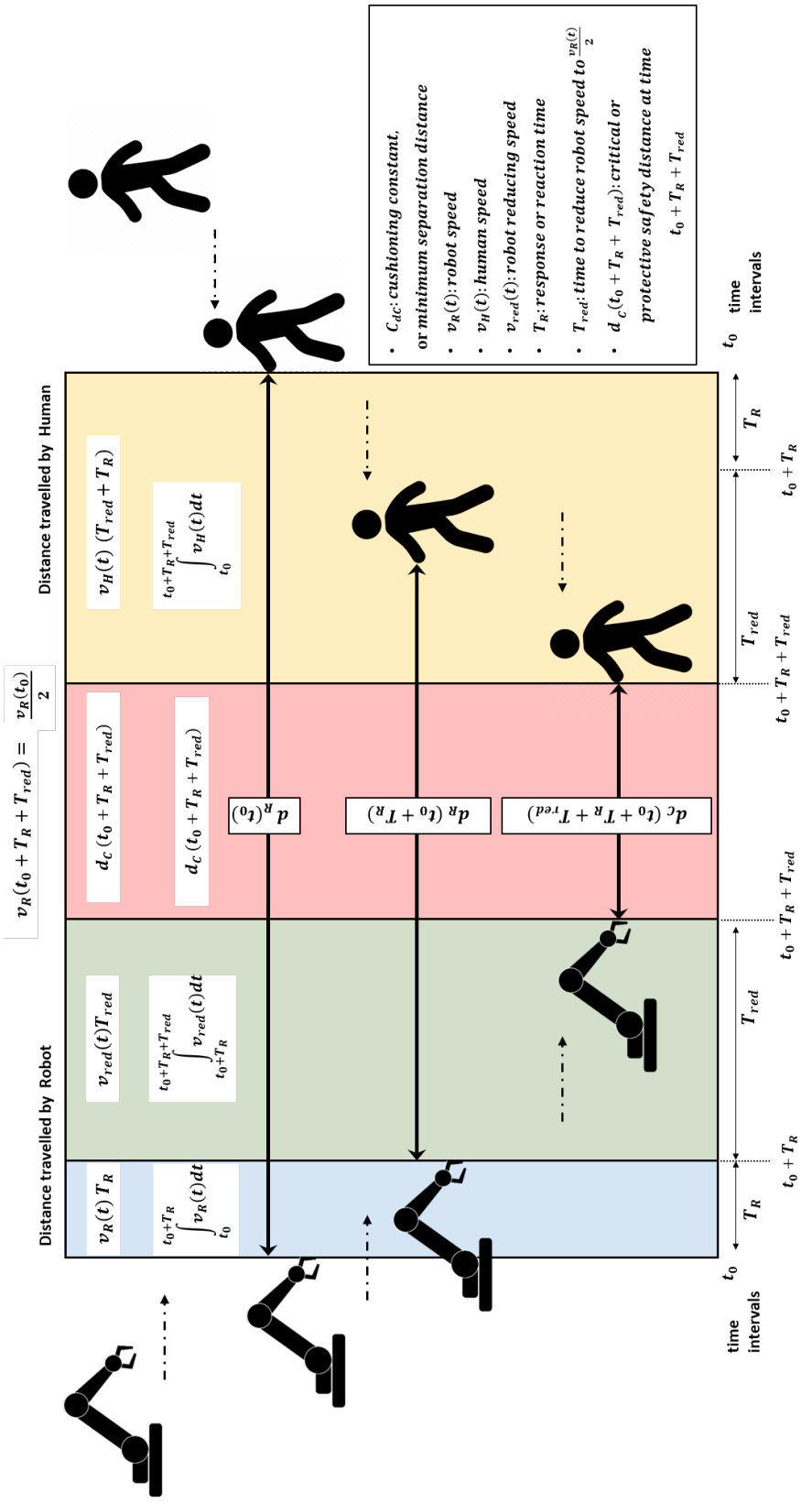


Figure 4.17: A time interval diagram representing the SSM formulation for defining the reduced safety distance between human and robot at time t_0.

during the reactive time interval (T_R) and the reduced time interval (T_{red}) as

$$d_R(t_0) \geq \underbrace{(v_o(t_0) + v_R(t_0))T_R}_{\text{change in relative dist. in } T_R} \underbrace{+ (v_o(t) + v_{red}(t))T_{red}}_{\text{change in relative dist. in } T_{red}} \underbrace{+ d_C(t_0 + T_R + T_{red})}_{\text{PSD}} \tag{4.34}$$

The form of representation in Eq.(4.34) is useful to see the relative distance change between the human and robot during the time intervals (T_R) and T_{red}. This is used later in the sections to define the contribution of the directed speeds (v_o) for PSD and RSD formulations for dynamic Tri-SSM based on ToF sensor arrays.

The next section describes the dynamic Tri-Modal SSM safety approaches implemented and evaluated using the aforementioned SSM formulations for ToF sensor arrays.

4.2.2 Dynamic Tri-Modal SSM Safety Approaches

In the Tri-SSM approach presented here, a robot action is taken when the distance safety index (DSI) calculated based on the minimum separation distance exceeds past the DSI calculated using PSD and RSD thresholds (refer Section 4.1.2.1). The PSD and RSD are updated based on the calculated directed speeds at every time step. A motion capture system (mocap) is integrated with digital-twin to control and localize the simulated human-avatar by tracking the human operator in the robot workspace. The mesh-mesh [1] minimum distance measured between the simulated robot and the human avatar is used as the ground truth for comparison of minimum distance measurements using the 8-node ToF sensor array prototype.

The Tri-SSM approaches are implemented for minimum separation distance calculated using the 8-node ToF sensor array prototype, the motion capture system, and the 2D Li-DAR. The minimum separation distances from different sensor modalities are described as follows.

(1) **Real ToF Sensor Array**(d_{Real}): The overall minimum separation distance d_{Real} calculated based on the distance measurements of the real hardware prototype of the 8-node

[1] A mesh is a collection of vertices, edges, and faces that describe the shape of a 3D object

ToF sensor arrays placed on the robot links. The details of the hardware and software of the prototype are detailed in chapter 6.

(2) **Simulated ToF Sensor Array**(d_{Sim}): The overall minimum distance d_{Sim} calculated based on the distance measurements of the simulated cone-shaped proximity sensors of the 8-node ToF sensor arrays placed on the robot links in the digital-twin simulation setup. This sensor modality was used in the first part of this Chapter in Section 4.1 to validate the use of ToF sensor arrays for a dynamic SSM safety approach.

(3) **Ideal Sensor Array**(d_{Ideal}): The overall minimum distance d_{Ideal} calculated by measuring the distance between the centers of the robot links to the closest point of the human-avatar mesh in the digital-twin setup. This minimum distance measurement represents the minimum human-robot separation distance for an ideal on-robot sensor array with no restrictions on thw sensors' field of view, loss of coverage due to blind-spots, or uncertainty of the sensor measurement. This minimum distance is used for comparison with the real ToF sensor arrays on the performance of dynamic Tri-SSM safety approaches.

(4) **Static 2D LiDAR** (d_{Lidar}): The minimum separation distance d_{Lidar} measured from the simulated 2D LiDAR proximity sensor placed at the base of the robot link and the human-avatar in a 2D plane parallel to the floor in the digital-twin setup. The 2D LiDAR is a conventionally used sensor modality in the industry, and is compared to the on-robot form of sensing.

(5) **Ground Truth**(d_{gt}): The absolute mesh-mesh minimum distance d_{GT} calculated in the digital-twin between all the points on the robot mesh and the human-avatar mesh. This distance is used as the ground truth for analyzing the accuracy of the minimum human-robot separation distance measurements.

These are three variations of the Tri-SSM safety configurations/approaches that are compared and evaluated. They are:

(a) **TriSSM-Vo**: This is a Tri-Modal dynamic SSM safety approach that takes into consideration the human/agent and robot directed speeds (v_o) for calculating the PSD and RSD thresholds according to the SSM formulations. Here, the human-agent/object is considered non-stationary, and the directed speed (v_o) is calculated as the projection of the relative velocity of the robot link and the human-agent/object on to the minimum distance vector (refer Section 4.1.2.2, Eq.(4.19)).

(b) **TriSSM-Vr**: This is a Tri-Modal dynamic SSM safety approach that only takes into

consideration the robot directed speed (v_r) for calculating the PSD and RSD thresholds according to the SSM formulations. The human-agent/object is considered stationary and the directed speed (v_r) is calculated as the projection of the robot-link velocity on to the minimum distance vector (refer Section 4.1.2.2, Eq.(4.19)).

(c) **TriSM**: This is a Tri-Modal SSM approach that does not consider the directed speeds and uses constant PSD and RSD thresholds. As the speeds are not considered, this approach is referred as Tri-Modal Separation Monitoring (TriSM) going forward. This is a form of dynamic SSM approach when used with on-robot ToF sensor arrays, and a static SSM while using a 2D LiDAR.

The three Tri-Modal SSM safety approaches **TriSSM-Vo**, **TriSSM-Vr**, and **TriSM**, based on the minimum separation distance calculated using four sensor modalities, **Real**, **Sim**ulated, **Ideal** and 2D **Lidar**, are evaluated and compared based on safety, performance, and productivity criteria. Thus, a total of 12 different SSM safety configurations are implemented and analyzed in this experiment.

In the following sections, the calculations of the minimum separation distance and directed speeds for the ToF-based sensor modalities (real, simulated and ideal) and 2D LiDAR are presented.

4.2.2.1 Minimum Distance and Directed Speeds using 2D LiDAR

A 2D scanning LiDAR proximity sensor simulated here is a model of a SICK 2D scanning LiDAR commonly used electronic safety device in industry. The area coverage of the 2D LiDAR is 180° degrees covering the shared collaborative workspace and a detection range of $3m$. The 2D LiDAR is placed at the base of the robot, as shown in Figure 4.18. The observations reported by the proximity sensor are in polar coordinates as $(o_{2D}(t), \theta_{2D}(t))$ with respect to the sensor frame. As the 2D LiDAR is placed at the base of the robot, the observations are interpreted from the base of the robot in a 2D plane parallel to the floor. Hence, the 2D position vector of the detected point on the human-agent in the workspace ($\boldsymbol{r_{o\,2D}(t)}$) at any given time ($t$) is given as

$$\boldsymbol{r_{o\,2D}(t)} = \begin{bmatrix} o_{2D}(t)\cos(\theta_{2D}) \\ o_{2D}(t)\sin(\theta_{2D}) \\ 0 \end{bmatrix} \tag{4.35}$$

The minimum separation distance using the 2D LiDAR is calculated between the projection of the end-effector (tool robot link l_j, $j = 3$) position, $\boldsymbol{r}_{\boldsymbol{l_3\,2D}}(\boldsymbol{q}) = [x_{l_3}\ y_{l_3}\ 0]^T$ for the robot pose ($\boldsymbol{q}$) and the detected point $\boldsymbol{r_{o\,2D}(t)}$ in the 2D sensing plane, as described in [46]. This can be written as

$$d_{2D_{min}}(t) = \|\boldsymbol{r_{o\,2D}(t)} - \boldsymbol{r_{l_3\,2D}}\|_2 \tag{4.36}$$

In the case when the robot joint pose (q) is not known, the minimum separation distance is measured from the circumference of the robot operating workspace of radius λ_{ows}, as shown in Figure 4.18. This is generally the case in industry for the implementation of static SSM. Hence, the minimum separation distance can be rewritten as

$$d_{2D_{min}}(t) = \begin{cases} o_{2D}(t) - \lambda_{ows} & \text{if } o_{2D}(t) > \lambda_{ows} \\ 0 & \text{otherwise} \end{cases} \tag{4.37}$$

This results in a conservative approximation of the minimum separation distance. In this experiment setup, the robot operating workspace, which is the distance of the end-effector/TCP from the base of the robot, when moving has a radius of $\lambda_{ows} = 0.82m$. In case of **TriSSM-Vo** based safety setup the velocity of the human-agent/object ($\dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t)$) is considered. The directed speed ($v_{o_{2D}}(t)$) based on the 2D LiDAR measurements using Eq.(4.19) is defined as

$$v_{o_{2D}}(t) = \max\left\{(\dot{\boldsymbol{r}}_{\boldsymbol{l_3\,2D}}(t) - \dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t))^T \frac{\boldsymbol{r_{o\,2D}}(t) - \boldsymbol{r_{l_3\,2D}}(t)}{\|\boldsymbol{r_{o\,2D}}(t) - \boldsymbol{r_{l_3\,2D}(t)}\|_2},\ 0\right\} \tag{4.38}$$

where $\dot{\boldsymbol{r}}_{\boldsymbol{l_3\,2D}}(t)$ and $\dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t)$ are the velocity vectors of the end-effector and the human/agent, respectively. The velocity vector of the detected human-agent/object $\dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t)$ can be written as

$$\dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t) = \frac{\boldsymbol{r_{o\,2D}(t)} - \boldsymbol{r_{o\,2D}}(t - \Delta t)}{\Delta t} \tag{4.39}$$

where Δt is the time step (sampling period) between the reported observations from the 2D LiDAR. The maximization indicates that the directed speeds are only considered when the robot is moving towards the human-agent/object. For **TriSSM-Vr**, the human-agent/object is considered stationary, which means $\|\dot{\boldsymbol{r}}_{\boldsymbol{o\,2D}}(t)\| = 0$. Hence, the directed speed ($v_{r_{2D}}(t)$)

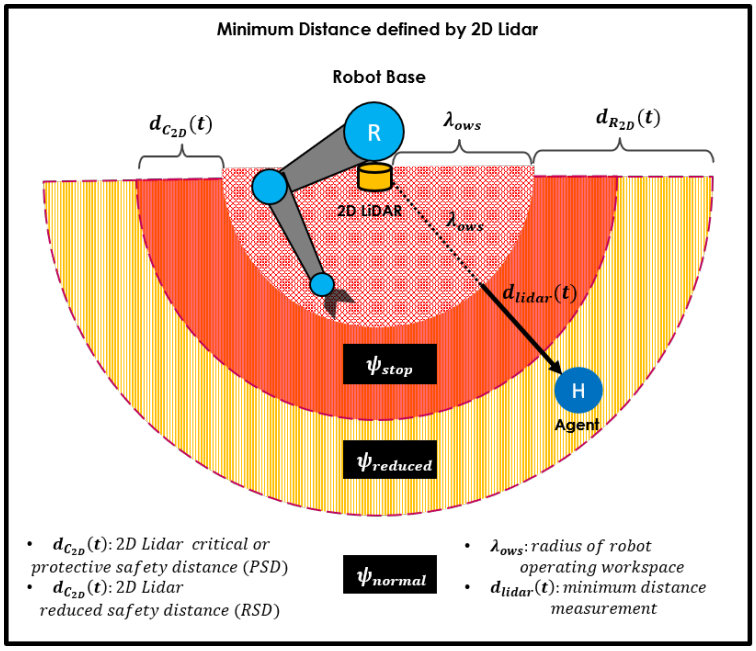


Figure 4.18: The protective/critical safety distance threshold $d_{C_{2D}}$ and the reduced safety distance threshold $d_{R_{2D}}$ defined for a 2D Lidar. The distance measurement by the 2D LiDAR of an agent in the robot workspace at time (t) is given as $d_{lidar}(t)$, which is measured from the circumference of the robot operating workspace of radius λ_{ows}.

during the **TriSSM-Vr** is written as

$$v_{r_{2D}}(t) = \max\left\{(\dot{\boldsymbol{r}}_{l_3\,2D}(t))^T \frac{\boldsymbol{r}_{o\,2D}(t) - \boldsymbol{r}_{l_3\,2D}(t)}{\|\boldsymbol{r}_{o\,2D}(t) - \boldsymbol{r}_{l_3\,2D}(t)\|_2},\ 0\right\} \tag{4.40}$$

In the following sections, the minimum distance calculations and directed speeds based on ToF Sensor arrays and motion capture is presented.

4.2.2.2 Minimum Distance and Directed Speeds using ToF sensor arrays

The minimum human-robot separation distance required for the implementation of dynamic SSM using ToF sensor arrays is calculated from the center of the robot link to the closest point on the human-agent and object. The minimum distance $d_{j\,min}(t)$ as measured by a ToF sensor array (S_j), between the closest human-agent/object point (r_{o_j}) and the

center of the robot links r_{l_j} can be written as

$$\begin{aligned} \boldsymbol{d}_{j\,min}(t) &= \boldsymbol{r}_{o_j}(\boldsymbol{t}) - \boldsymbol{r}_{l_j}(\boldsymbol{t}) \\ \|\boldsymbol{d}_{j\,min}(t)\|_2 \hat{\boldsymbol{d}}_{j\,min}(\boldsymbol{t}) &= \|\boldsymbol{r}_{o_j}(\boldsymbol{t}) - \boldsymbol{r}_{l_j}(\boldsymbol{t})\|_2 \hat{\boldsymbol{d}}_{j\,min}(\boldsymbol{t}) \end{aligned} \tag{4.41}$$

In the case of an **Ideal** ToF sensor array setup, the digital-twin is used to determine the position vector r_{o_j} of the closest point on the mesh of the human-avatar mimicking the motion and position of the human-operator in the physical world using the motion capture setup. This form of sensing uses the motion capture setup to model the ideal behavior of an on-robot sensor placed at the center of the robot links.

In case of the **Real** and **Simulated** 8-node ToF sensor array prototype, the minimum separation distance is calculated based on Eq.(4.16a) as

$$\begin{aligned} \|\boldsymbol{d}_{j\,min}(t)\|_2 \equiv d_{j\,min}(t) &= \lambda_j + \min_i \left\{d_{ji}(t)\right\} \\ k_j(t) &= \underset{i}{\operatorname{argmin}} \left\{d_{ji}(t)\right\} \end{aligned}$$

To incorporate the influence of human-robot relative velocity in calculations of PSD and RSD thresholds, the directed speeds v_{o_j} of all robot links ($where\, j = \{base, elbow, tool\}$) moving towards the human-agent/object are considered. The component of the human-robot relative velocity on the minimum distance vector v_{o_j} for the $j-$th robot link was defined previously in Eq.(4.19) as

$$\begin{aligned} v_{o_j}(t) &= \left(\dot{\boldsymbol{r}}_{l_j}(t) - \dot{\boldsymbol{r}}_{o_j}(t)\right)^T \left(\frac{\boldsymbol{r}_{o_j}(\boldsymbol{t}) - \boldsymbol{r}_{l_j}(t)}{\|\boldsymbol{r}_{o_j}(\boldsymbol{t}) - \boldsymbol{r}_{l_j}(\boldsymbol{t})\|_2}\right) \\ &= \left(\dot{\boldsymbol{r}}_{l_j}(t) - \dot{\boldsymbol{r}}_{o_j}(t)\right)^T \hat{\boldsymbol{d}}_{j\,min}(\boldsymbol{t}) \\ &= \left(\frac{\boldsymbol{r}_{l_j}(t) - \boldsymbol{r}_{l_j}(t-\Delta t)}{\Delta t} - \frac{\boldsymbol{r}_{o_j}(t) - \boldsymbol{r}_{o_j}(t-\Delta t)}{\Delta t}\right)^T \hat{\boldsymbol{d}}_{j\,min}(\boldsymbol{t}) \end{aligned} \tag{4.42}$$

After some algebraic manipulation, the above equation can be rewritten as

$$
\begin{aligned}
v_{o_j}(t) &= \left(\frac{\boldsymbol{r_{o_j}}(t-\Delta t) - \boldsymbol{r_{l_j}}(t-\Delta t)}{\Delta t}\right)^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}) - \left(\frac{\boldsymbol{r_{o_j}}(t) - \boldsymbol{r_{l_j}}(t)}{\Delta t}\right)^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}) \\
&= \left(\frac{\boldsymbol{d_{j\ min}}(t-\Delta t)}{\Delta t}\right)^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}) - \left(\frac{\boldsymbol{d_{j\ min}}(t)}{\Delta t}\right)^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}) \\
&= \left(\frac{\|\boldsymbol{d_{j\ min}}(t-\Delta t)\|\ (\hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t-\Delta t})^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}))}{\Delta t}\right) - \left(\frac{\|\boldsymbol{d_{j\ min}}(t)\|(\hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t})^T \hat{\boldsymbol{d}}_{\boldsymbol{j\ min}}(\boldsymbol{t}))}{\Delta t}\right) \\
&= \left(\frac{\|\boldsymbol{d_{j\ min}}(t-\Delta t)\|\ \cos(\theta_j(t))}{\Delta t}\right) - \left(\frac{\|\boldsymbol{d_{j\ min}}(t)\|}{\Delta t}\right) \\
&= \frac{\left(d_{j\,min}(t-\Delta t)\cos(\theta_j(t)) - d_{j\,min}(t)\right)}{\Delta t}
\end{aligned}
\tag{4.43}
$$

where $\theta_j(t)$ is the angle between the directions of the minimum distance vectors at time (t) and $(t-\Delta t)$. In other words, it is the difference of the orientation of the ToF sensor nodes that reported the minimum distance (which are sensor nodes indices $k_j(t)$ and $k_j(t-\Delta t)$ for the ToF sensor array S_j). In the case where the distance measurement sampling rate is fast enough or the angle between the sensors that reported the minimum distance at times (t) and $(t-\Delta t)$ is very small ($\lim_{\theta_j \to 0} cos(\theta_j) = 1$), the directed speed $v_{o_j}(t)$ can be approximated as

$$
\lim_{\theta_j(t)\to 0} v_{o_j}(t) = \frac{\left(d_{j\,min}(t-\Delta t) - d_{j\,min}(t)\right)}{\Delta t} \tag{4.44}
$$

The above equation is the rate of change in the measurement of minimum separation distance by the ToF sensor array (S_j) placed at the center of the $j-$the robot link. Two scenarios need to be carefully considered while using the above formulation. First is when the human-agent/object enters or exits the detection range, and second is when a new closest point is detected where $\theta_j(t)$ is substantial. In both these cases, the rate of change of the perceived minimum distance will be large, resulting in an erroneous measurement of the directed speed. This is mitigated by using an exponential smoothing filter.

The directed speed for **TriSSM-Vo** safety approach for all ToF and motion capture-based sensor modalities $v_{o_j\,Real,Sim,Ideal}(t)$ can be defined as

$$v_{o_j\,Real,Sim,Ideal}(t) = \max\{v_{o_j},\ 0\} \tag{4.45}$$

As mentioned previously, the maximization indicates the consideration of only positive directed speeds indicating that the robot and human-agent are moving towards each other.

Similarly, the directed speed for **TriSSM-Vr** safety approach for all ToF-based sensor modalities $v_{r_j\,Real,Sim,Ideal}(t)$, where the human-agent/object is considered stationary can be defined as

$$v_{r_j\,Real,Ideal,Sim}(t) = \max\left\{(\dot{\boldsymbol{r}}_{\boldsymbol{l_j}}(t))^T\hat{\boldsymbol{d}}_{\boldsymbol{j\,min}}(\boldsymbol{t}),\ 0\right\} \tag{4.46}$$

Next, these directed speeds are used in defining the SSM formulations for PSD and RSD for ToF-based sensor arrays. These formulations are shown in the following section.

4.2.2.3 SSM Formulation for ToF sensor arrays

In this research for dynamic SSM using ToF sensor arrays, the danger/safety is quantified independently for each robot's links. For the ToF sensor arrays placed at the robot link centers l_j (where $j = \{base,\ elbow,\ tool\}$) and separation distance $d_{j\,min}(t_0)$ at a given time (t_0), perceived by the ToF sensor array S_j, a PSD $d_{C_j}(t_0)$ can be calculated with respect to the centers of the robot links (l_j) as

$$\begin{aligned} d_{C_j}(t_0) \geq \underbrace{(v_{o_j}(t_0)T_R + v_{o_j}(t_0)T_{stop})}_{\text{human distance}}\ \underbrace{+\,\|\dot{\boldsymbol{r}}_{\boldsymbol{l_j}}(t_0)\|T_R}_{\text{robot reaction distance}} \\ \underbrace{+\,B}_{\text{robot braking distance}}\ \underbrace{+\,C_{dC}}_{\text{cushioning constant}} \end{aligned} \tag{4.47}$$

where $\|\dot{\boldsymbol{r}}_{\boldsymbol{l_j}}(t_0)\|$ is the magnitude of the velocity of the $j-$th robot link center at time (t_0).

Similarly, for a dynamic Tri-SSM safety setup, the RSD for the $j-$th robot link based

on the Eq.(4.34) is defined as

$$\begin{aligned} d_{R_j}(t_0) \geq & \underbrace{(v_{o_j}(t_0) + \dot{\boldsymbol{r}}_{l_j}(t_0)\|)T_R}_{\text{change in relative dist. in } T_R} \underbrace{+ (v_{o_j}(t) + v_{red}(t))T_{red}}_{\text{change in relative dist. in } T_{red}} \\ & \underbrace{+ d_{C_j}(t_0 + T_R + T_{red})}_{\text{PSD}} \\ d_{R_j}(t_0) \geq & \underbrace{d^*_{R_j}(t_0)}_{\text{directed speed influence}} + d_{C_j}(t_0 + T_R + T_{red}) \end{aligned} \tag{4.48}$$

where $d_{C_j}(t_0 + T_R + T_{red})$ according to Eq.(4.32) is defined as

$$\begin{aligned} d_{C_j}(t_0 + T_R + T_{red}) \geq & \underbrace{(v_{o_j}(t_0)T_R + v_{o_j}(t_0)T_{stop})}_{\text{human distance}} \underbrace{+ \frac{\|\dot{\boldsymbol{r}}_{l_j}(t_0)\|T_R}{2}}_{\text{robot reaction distance}} \\ & \underbrace{+ B}_{\text{robot braking distance}} \underbrace{+ C_{dC}}_{\text{cushioning constant}} \\ d_{C_j}(t_0 + T_R + T_{red}) \geq & \underbrace{d^*_{C_j}(t_0)}_{\text{directed speed influence}} + \frac{\|\dot{\boldsymbol{r}}_{l_j}(t_0)\|T_R}{2} + B + C_{dC} \end{aligned} \tag{4.49}$$

where directed speed influence on the critical ($d^*_C(t_0)$) and reduced ($d^*_R(t_0)$) distance thresholds can be calculated by estimating the area under the curve of graphs shown in Figure 4.19. The assumption made is that the directed speed reduces linearly. Assuming linear deceleration is the simplest approach for calculating the contribution of human and robot velocities for PSD, as described in [13, 46]. In actuality, a more accurate and less conservative representation of the rate of change in the directed speeds is by using a non-linear deceleration profiles, as shown in [46, 85]. For ease of notations in this section, the PSD and RSD are implied to be calculated for a single $j-$th robot link.

The time intervals $\{T_{stop}, T_{red}\}$ are defined as the time taken for the robot to stop and slow down to half the operating speed, respectively. The interval of the reaction time (T_R) is the sum of the delay by the robot to start executing an action following a command, the processing time to determine the directed speeds and minimum distances, and the time taken to trigger the robot action. Let v_o be the human-robot directed speed and its corresponding maximum value be v_{omax}, which is the maximum relative speed when the human

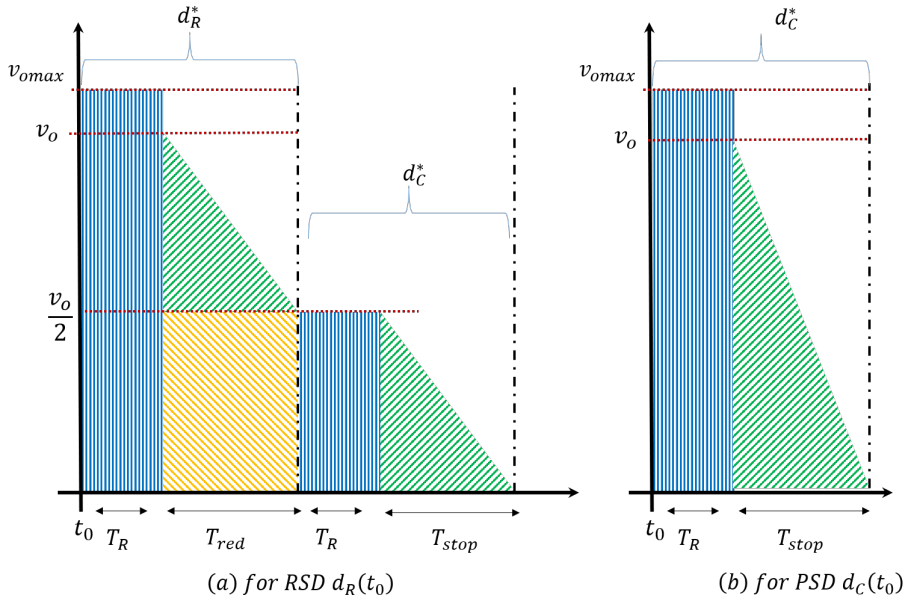


Figure 4.19: Graph representing the influence of directed speed (v_o) of a robot and a human-agent/obstacle in the calculation of safety distance thresholds RSD and PSD, at time (t_0), for TriSSM-Vo and TriSSM-Vr safety configuration.

and robot are moving towards each other. During the reaction time interval T_R, it is assumed that the robot is moving at the maximum directed speed v_{omax}. During the interval of T_{red}, as the robot reduces its operating speed, an assumption is made that the directed speed (v_o) reduces linearly.

Thus, the formulation of $d_R(t_0)$ at time (t_0) taking into account the stopping distance PSD $d_C(t_0 + T_R + T_{red})$ at time $(t_0 + T_R + T_{red})$ (essentially the time taken by the robot to come to a complete stop after reducing the speed) can be defined as

$$d_R(t_0) \geq \underbrace{(v_{omax}T_R) + v_o(t_0)(\frac{3}{4}T_{red})}_{d_R^*} + d_C(t_0 + T_R + T_{red}) \tag{4.50a}$$

where

$$d_C(t_0 + T_R + T_{red}) \geq \underbrace{(\frac{v_o(t_0)T_R}{2}) + (\frac{v_o(t_0)T_{stop}}{4})}_{d_C^*} + \frac{\|\dot{\boldsymbol{r}}_I(t_0)\|T_R}{2} + B + C_{dC}$$

To avoid sudden increases in robot operation speeds when transitioning from $\{\psi_{stop} = 0, \psi_{reduced} = 1\}$ to $\psi_{normal} = 2$, a recovery buffer distance R_{buffer} is added to the above equation of $d_R(t)$ as

$$d_R(t_0) \geq \underbrace{(v_{omax}T_R) + v_o(t_0)(\frac{3}{4}T_{red})}_{d_R^*} + R_{buffer}(t_0) + d_C(t_0 + T_R + T_{red})$$

$$\text{where } R_{buffer}(t_0) = (\psi_{normal} - \psi(t_0))(\frac{C_{rec}}{2})$$

where C_{rec} is a recovery constant (akin to C, the cushioning constant).

For **TriSSM-Vr** based configuration, v_o and v_{omax} are replaced with the robot directed speeds v_r and v_{rmax}. This formulation is used for all sensor modalities which are ToF, motion capture and 2D LiDAR.

Summary

To summarize, the final formulations for calculating PSD ($d_{C_j}(t_0)$) and RSD ($d_{R_j}(t_0)$) at any given time (t_0) for the $j-$th robot link moving towards a human agent/object at a directed speed $v_{o_j}(t_0)$ used for the implementation of TriSSM safety approaches for the ToF and motion capture-based sensor modalities (*Real*, *Sim* and *Ideal*) are

$$\begin{aligned} \text{RSD, } d_{R_j}(t_0) \geq (v_{omax}T_R) + v_{o_j}(t_0)(\frac{3}{4}T_{red}) + (\psi_{normal} - \psi_j(t_0))(\frac{C_{rec}}{2}) \\ + v_{o_j}(t_0)(\frac{T_R}{2} + \frac{T_{stop}}{4}) + \frac{\|\dot{\boldsymbol{r}}_{l_j}(t_0)\|T_R}{2} + B + C_{dC} \end{aligned} \tag{4.52a}$$

$$\begin{aligned} \text{PSD, } d_{C_j}(t_0) \geq (v_{omax}T_R + \frac{v_{o_j}(t_0)T_{stop}}{2}) \\ + \|\dot{\boldsymbol{r}}_{l_j}(t_0)\|T_R + B + C_{dC} \end{aligned} \tag{4.52b}$$

For the 2D LiDAR, the robot link considered for calculation of PSD and RSD is the end-effector/TCP ($j = 3(tool)$).

Next, we present the methodology for filtering the distance measurements of real ToF sensor arrays placed on the robot corresponding to the restricted workspace such as tables, floors, and the robot self.

4.2.3 Self Occlusion Check

The ToF sensor nodes in each array will also detect other robot links and other objects in the work-space. In order to ignore these sensor readings, the sensor detection is modeled as shown in Figure 4.20 using ray-casting in a physics engine [75]. Let the collection of objects that are stationary and belong to the restricted workspace of the robot be $\boldsymbol{W_{restricted}}$ and that of the robot links are attached to the links be $\boldsymbol{W_{robot}}$. For a i-th sensor on ToF sensor array S_j gives a sensor measurement d_{ji} and sensor accuracy σ, the ray-cast results in a distance d_{ray_j} of the intersection of object O_k in the workspace. The self-occlusion binary mask, m_{ji}, can be written as follows:

$$m_{ji} = \begin{cases} 0, & \text{if } \{O_k \in (\boldsymbol{W_{restricted}} \cup \boldsymbol{W_{robot}}\} \ \ \& \ \ d_{ray_j} \in [d_{ji} \pm 3\sigma] \\ 1, & \text{otherwise} \end{cases} \tag{4.53a}$$

$$\widehat{d_{ji}} = d_{ji} \cdot m_{ji} \tag{4.53b}$$

This approach is very similar to collision masking in a physics engine [29, 75, 76]. The operation of ray-casting can be expensive if the rays of intersection are displayed. Still, if the process is running as an independent process, the average execution time taken is $\approx 2ms$ or $500Hz$ for 600 rays. This rate is enough to mask the ToF sensors reading data at $27Hz - 30Hz$. The faster execution is one of the main advantages of using an external physics engine instead of the digital-twin in V-REP.

4.2.4 Distance Safety Index Algorithm

In the Tri-SSM approach presented here, a robot action is taken when the distance safety index (DSI) calculated based on the minimum separation distance exceeds past the DSI calculated using PSD and RSD thresholds (refer Section 4.1.2.1). The PSD and RSD are updated based on the calculated directed speeds at every time step. The minimum allowable(or acceptable) distance, $d_{min_{thresh}} = C_{dC}$, is used for normalization of DSI (see Eq. (4.10)). Thus, the normalized DSI for the minimum separation distance $d_{j\,min}(t)$, the PSD $d_{C_j}(t)$ and the RSD $d_{R_j}(t)$ for the j-th robot link (where $j = \{1(base),\ 2(elbow),\ 3(tool)\}$)

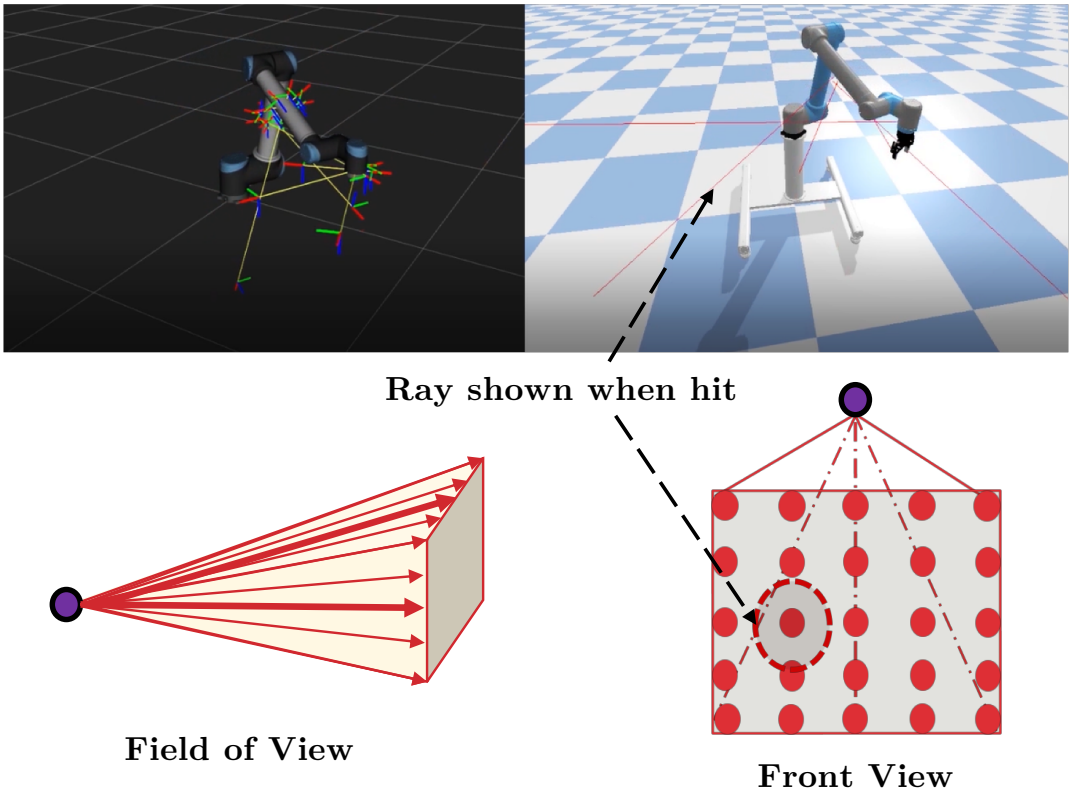


Figure 4.20: Self Occlusion Check using ray casting of all ToF sensors using PyBullet physics engine. The lower half of the figure shows the perspective and front view of the FoV of the sensor depicted as a point source.

are defined as

$$\widetilde{dsi_j}(t) = \left(\frac{C_{dC}}{d_{j\,min}(t)}\right)^2 \tag{4.54a}$$

$$\widetilde{dsi_{C_j}}(t) = \left(\frac{C_{dC}}{d_{C_j}(t)}\right)^2 \tag{4.54b}$$

$$\widetilde{dsi_{R_j}}(t) = \left(\frac{C_{dC}}{d_{R_j}(t)}\right)^2 \tag{4.54c}$$

A robot action/operation mode ψ_j is determined for each robot link $j \in \{base, elbow, tool\}$, and the action/operation mode reported by the link with the greatest danger is chosen. Thus, the overall robot action/operation mode determined is $\psi = \min_j\{(\psi_j\}$, as detailed in Algorithm 5. The data-flow diagram of the Algorithm 5 is shown in Figure 4.21. Similar to the overall safety index Algorithm 4, an exponential smoothing filter with varying filter

coefficients based on the DSI change is implemented.

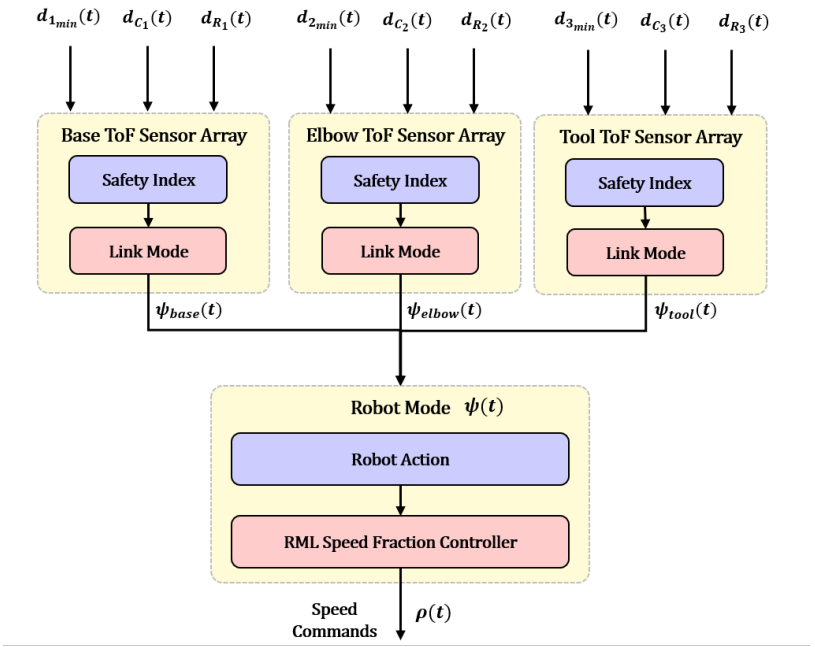


Figure 4.21: A data flow diagram for the DSI Algorithm 5 for dynamic Tri-SSM used for determining the robot operation modes.

The robot movement speed is controlled using a Reflexxes Motion Library (RML) [81] based on-line controller. This controller generates non-linear profiles of the rate of speed-fraction change and sends movement commands to the robot at $125Hz$. The primary purpose of this controller is to impose time-critical constraints that ensure the robot speed reduction within a fixed time interval. It also results in a smoother transition and robust control of the robot motion and its operating speed. The interface for the control of the robot is discussed further in the following section.

Algorithm 5: Distance Safety Index-based Tri-SSM Algorithm

Data: exponential filter weights ($\alpha_D = 0.8$, $\alpha_I = 0.3$),
speed fraction ($\rho_{normal} = 1$, $\rho_{reduce} = 0.5$, $\rho_{stop} = 0$)

Input : previous DSI at time $(t - \Delta t)$, $\widetilde{dsi}_j(t - \Delta t)$,
minimum distance $d_{j\,min}(t)$, PSD $d_{C_j}(t)$ and RSD $d_{R_j}(t)$,
$\forall\ j \in \{base, elbow, tool\}$

Output: robot action ψ

```
    /* robot action for all links                                          */
1   for j ∈ {1(base), 2(elbow), 3(tool)} do
        /* calculate DSI                                                   */
2       d̃si_j(t) ← calculateDSI(d_{j min}(t))
3       d̃si_{R_j}(t) ← calculateDSI(d_{R_j}(t))
4       d̃si_{C_j}(t) ← calculateDSI(d_{C_j}(t))
        /* variable exponential filter for smoothening DSI                 */
5       if d̃si_j(t) < d̃si_j(t − Δt) then
6           α = α_I
7       else
8           if d̃si_j(t) > d̃si_{C_j}(t) then
                /* highest danger, immediate action , no filtering
                                                                            */
9               α = 1
10          else
11              α = α_D
12          end
13      end
14      d̃si_j(t) = (α)d̃si_j(t) + (1 − α)d̃si_j(t − Δt)
15      d̃si_j(t − Δt) ← d̃si_j(t)
        /* determin robot action according to robot link j                 */
16      if d̃si_j(t) < d̃si_{R_j}(t) then
            /* move robot normally                                         */
17          robot action ψ_j ← ψ_normal
18      else if d̃si_{R_j}(t) ≤ d̃si_j(t) < d̃si_{C_j}(t) then
            /* reduce the robot speed                                      */
19          robot action ψ_j ← ψ_reduce
20      else
            /* stop the robot                                              */
21          robot action ψ_j ← ψ_stop
22      end
23  end
```

```
24    ψ = min_j{ψ_j} ;
      /* take robot action                                   */
25    if ψ is ψ_normal then
          /* move robot normally                             */
26        setRMLspeedFraction(ρ_normal);
27    else if ψ is ψ_reduced then
          /* reduce the robot speed                          */
28        setRMLspeedFraction(ρ_reduce);
29    else
          /* stop the robot                                  */
30        setRMLspeedFraction(ρ_stop);
```

4.2.4.1 Robot Interface

It is vital to monitor and command a robot in real-time to successfully implement dynamic SSM or any safety setup due to its time-constraint nature. Moreover, the robot desired speeds cannot be instantly achieved, as it could generate significant jerky motion and potentially cause damage to the robot actuators. Hence, a robust robot communication interface and a controller for the transition of robot speeds to reduce jerk is implemented.

The UR10 robot provides its internal state at $125Hz$ through a robust communication protocol called RTDE [11] and the robot-state information is assumed to be accurate. As the intended purpose of this application is not to control the robot trajectory but to change the joint speed fraction ρ, this interface is used to vary the speed fraction through the control provided by the UR10 interface. A network-based real-time interface and RTDE communication is used to update the speed fraction at the rate of $125Hz$. For the implementation of SSM, the robot state information used are joint angles, joint velocities and joint accelerations ($\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \ddot{\boldsymbol{q}}(t)$), the robot end-effector/TCP velocity $V_r(t)$ and digital IOs pin states.

For reducing jerks and smoother transition when changing safety states (normal, reduced, and stop), an on-line controller using the Reflexxes Motion Library (RML) TypeII [81] to control the rate of change of the joint speed fraction ρ. This ensures that sudden changes in robot operating modes and action transitions do not abruptly change the

robot speeds. This speed-fraction controller is used for on-line generations of acceleration/deceleration profiles for achieving a target speed fraction ρ_{target}. This controller generates robot speed commands based on the profiles at $125Hz$ with values that ensure reduction as well as an increase in robot speeds under fixed time-critical constraints.

The acceleration/deceleration values are generated under time-constraints of the maximum time allowed to transition between the previous ($\psi(t - \Delta t)$) and current desired robot modes ($\psi(t)$). A time-constraint, which is the maximum time allowed to transition between the robot modes, $T_{prev,curr}$, is defined. For example, given a transition between previous $\psi_{normal} = 2$ and current $\psi_{reduced} = 1$, a series of speed fraction control commands are generated by the RML controller that ensures reduction of speed under a fixed time constraint T_{21}. In a more conservative robot behavior, the transition from a dangerous mode ($\psi_{stop} = 0$) to safer mode ($\psi_{normal} = 2$), the desired robot behavior is when the robot slowly and cautiously increases its speed.

4.2.5 Experiment Setup and Evaluation Criteria

The initial experiment was done to test the performance of the 8-Node ToF sensor array prototype. A distance-based dynamic Tri-SSM (refer Section 4.1.2.3 D. , Algorithm 3) was implemented using the real 8-node ToF sensor arrays, as shown in Figure 4.22. For the real 8-node ToF sensor array prototype, the information from the digital-twin is not required for performing SSM.

An experiment with an HRC task was performed to compare the safety, performance, and productivity of the twelve different combinations of the Tri-Modal SSM safety configurations as shown in Figure 4.23. A schematic of the experiment and validation setup is shown in Figure 4.24.

The experiment setup is a generic robot pick and place task of placing ten products in a box. It is the same setup used during the implementation of dynamic SSM using simulated ToF sensors, as described in Section 4.1.3. In order to validate and compare the dynamic SSM safety configurations for proximity-based sensors, it requires the use of controlled physical avatars that can move in repeatable trajectories to maintain the same human interaction. In this setup, the human path is fixed as '∞' shaped path overlapping the robot's operating workspace, as shown in Figure 4.24. In order to validate the system,

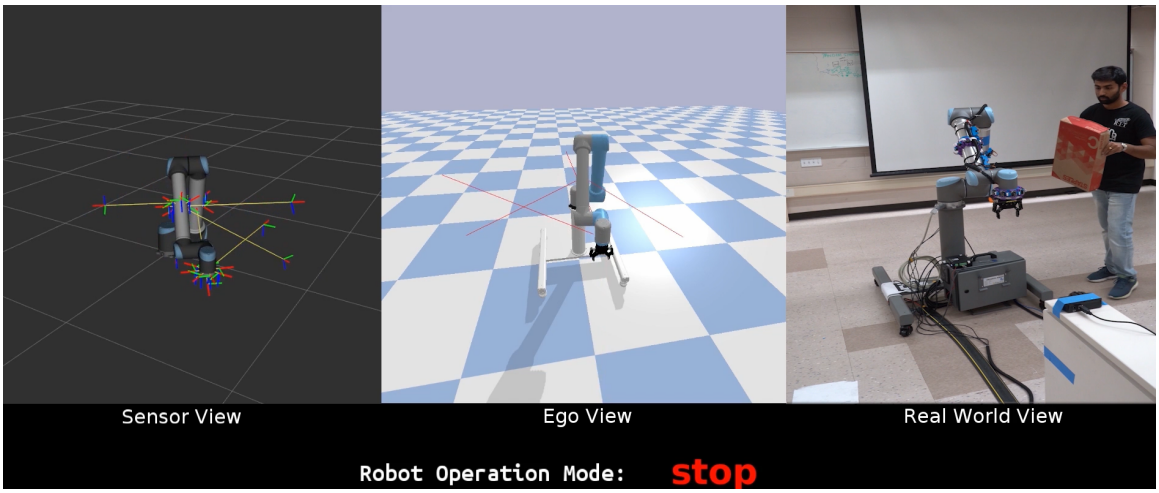


Figure 4.22: An implementation of distance-based Tri-Modal Dynamic SSM for a UR10 robot with 8-node ToF sensor array prototypes placed at the centers of robot links Video Link.

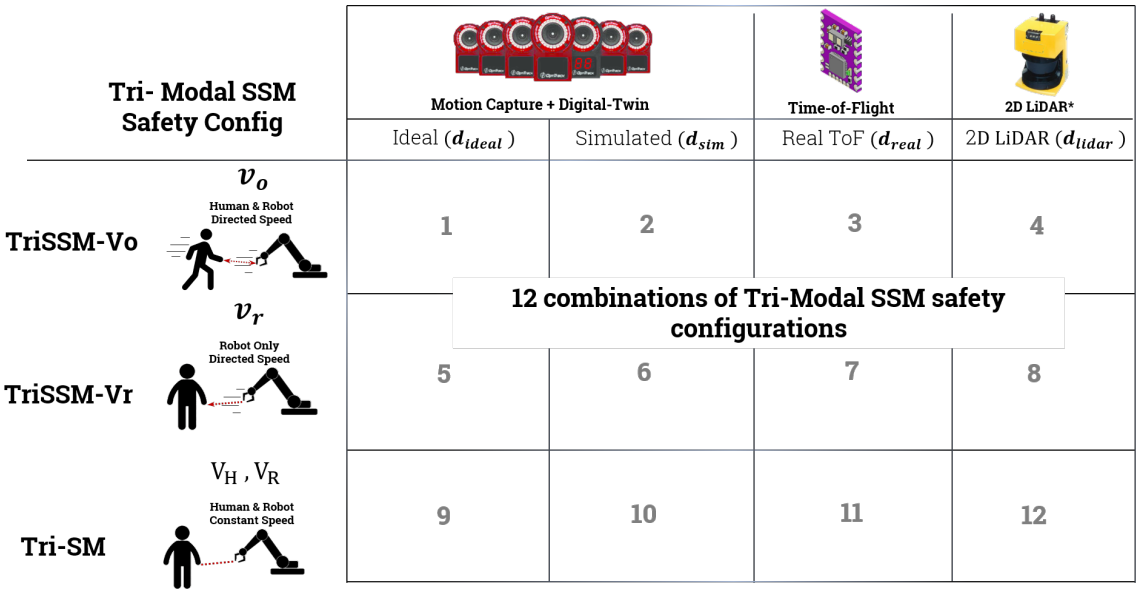


Figure 4.23: A grid showing the twelve combinations of the Tri-Modal SSM safety configurations evaluated.

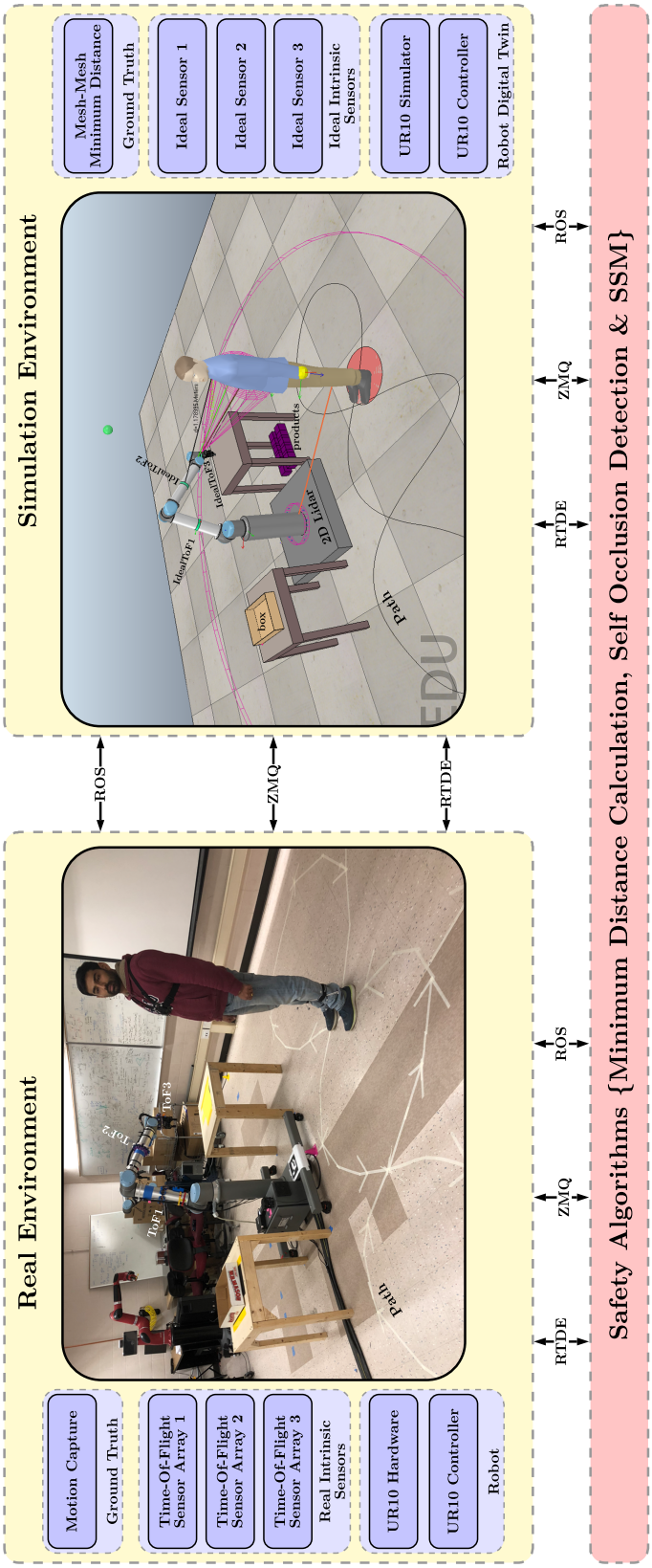


Figure 4.24: A schematic of the system used to implement, validate and test the dynamic Tri-SSM safety configuration(s). The transport layer for communication between different subsystems such as the real and virtual environments is built using ZeroMQ, RTDE and ROS. Video Link

the V-REP simulation environment [29, 76] is used to generate a *digital-twin* [114] of the experiment where a human-avatar mimics the motion of the human tracked using a motion-capture system. The robot pose and movements in V-REP are updated based on the reported states and movement of the robot in the real world. A similar approach has been used in [55]. In the experiment for a given SSM safety configuration, the human motion, the states reported by the robot, and performance of the dynamic SSM safety configurations are recorded. The human skeleton movements as reported by the motion capture are also recorded. For the motion capture-based Ideal and 2D LiDAR-based setups, the robot is made to *hallucinate* the presence of the human in the real world by moving the human-avatar in V-REP according to the human recorded movements. This ensures the human interaction across all minimum distance calculation approaches is the same. This is done at $\approx 120Hz - 125Hz$ based on the information provided by the robot and the motion capture system.

It should be noted that V-REP is used strictly for validation of the *Real* ToF sensor array hardware to compare to the ground truth (mesh-mesh distance) and *Ideal* minimum distance(s). It can be seen in the schematic shown in Figure 4.24, that the real and the simulation (digital-twin) environments are independent for implementation of dynamic SSM. The human-avatar in the digital-twin is controlled based on the human-operator movement in the real world based on the tracking information from the motion-capture setup. The human-robot minimum distance measurements, d_{Sim}, d_{Ideal} and d_{LiDAR}, are provided using the digital-twin by calculating the distances between the simulated robot and the human-avatar. For robust and real-time communication of information between the motion-capture system, the robot, the sensors and the digital-twin setup, a ZeroMQ [106] based communication layer is implemented. Robot Operating System (ROS) [107] is used for storing the information during the TriSSM experiment in ROSBAG containers.

It is important to evaluate the accuracy of the minimum distance calculation and compare it with the ground truth d_{gt} (the mesh-mesh minimum distance) as this provides the error in the minimum distance calculation. The metrics for Safety, Performance and Productivity, as shown in Table 4.4 is calculated for all SSM configurations (*Vo, Vr, SM*) and minimum distance calculation approaches (*Real, Ideal, Lidar*).

One of the safety criteria is the change in robot end-effector/TCP velocity at the time the robot action to stop (ψ_{stop}) is triggered. This metric indicates the anticipatory and reactive

nature of the Tri-SSM safety setup. A smaller magnitude of change in velocity when the stop event occurs indicates smoother joint speed transitions and less jerky motion. The average human-separation distance during the HRC task is an indicator of how comfortable the human was sharing the robot workspace. In other words, if the human operator felt comfortable sharing the workspace with the robot, they would work in closer proximity and not keep a distance. The last criteria is a safety metric as described in [85], can be used to define the least safe moment during the experiment. The overall safety is considered less when the velocity of the end-effector (considered the fastest point on the robot) is high and the minimum separation distance is less. The formulation of the safety metric is defined as

$$Safety\,Metric = \frac{d_{gt}^2}{v_{TCP}} \tag{4.55}$$

where d_{gt} is the ground truth minimum distance (motion-capture) and $v_{TCP} = \|\dot{\boldsymbol{r}}_{\boldsymbol{t_3}}\|_2$ is the robot end-effector/TCP velocity. The performance of an HRC setup is dependent on the objective of task. In this case, a dynamic SSM setup to increase the human-operator safety is evaluated. The robot actions during the dynamic SSM safety setup are time-constrained. Hence, an average stopping and reducing time are the performance metrics used for evaluation. The faster reaction times (smaller value in seconds) would suggest the anticipatory nature of the safety setup and the robustness of the robot actions taken under the desired time-constraints. Another important performance metric is the robot reaction time to stop. This is defined as the time taken by the robot to come to a complete halt prior to the minimum distance measured goes below the minimum acceptable distance threshold or the cushioning constant C_{dC}. This time should be negative indicating that the robot did not move when the human-robot minimum distance was less than the minimum allowable human-robot separation distance.

A desired TriSSM safety configuration for an HRC task is the one that optimizes productivity while ensuring human safety. The productivity of an HRC task for SSM was defined in [85] as :

$$Productivity = \frac{t_{NoHRI}}{t_{HRI}} \tag{4.56}$$

where t_{NoHRI} and t_{HRI} are the time taken to complete the given task with no human interaction and with human interaction, respectively.

Table 4.4: An evaluation metric for the proposed system.

CRITERIA	METRIC DESCRIPTION
Safety	•Robot TCP Velocity Change at Stop Event •Average human-robot separation distance •Safety Metric as per Eq. 4.55
Performance	•Average Stopping & Reduced Time •Average Reaction Time
Productivity	•Time taken to complete the task with HRI as per Eq. 4.56

Next, we present the results and observations of the TriSSM approaches evaluated based on the evaluation criteria mentioned above.

4.2.6 Results and Observation

There were three variations of the safety approaches **TriSSM-Vo, TriSSM-Vr** and **Tri-SM** implemented using the real 8-node ToF sensor array prototype (d_{Real}), the simulated ToF sensors in digital-twin and motion capture (d_{Sim}), the ideal on-robot sensor in the digital-twin and motion capture (d_{Ideal}) and, the simulated 2D LiDAR (d_{Lidar}). Thus, there were in total of twelve different safety configurations implemented and evaluated. There were five trials for each dynamic SSM safety setup experiment. Here, we present the summarized results of the observations and evaluations of the configurations.

The comparison of the all the safety approaches is mainly between the real 8-node ToF sensor array(d_{Real}), the ideal sensor (d_{Ideal}) and the 2D LiDAR (d_{Lidar}). The simulated sensors (d_{Sim}) can be considered the same form of measurement as the motion-capture with the sensing coverage of the real 8-node ToF sensor array.

The experiments were performed with the parameters given in Table 4.5. The time interval $T_{stop} = 0.4s$ is the maximum time taken by the UR10 robot to come to a complete stop when moving at the maximum speed carrying the maximum payload when fully extended. This time interval was defined in the safety function for UR10 robots [59]. In this setup, using the Reflexxes Motion Library (RML) a deceleration profile for speed reduction for the interval of $T_{stop} = 0.4s$ and $T_{red} = 0.4s$ is generated. The time constraints for the generation of robot speed commands and the time taken to execute them are taken into

consideration to determine T_{stop} and T_{red}. This ensures smoother and controlled transitions between robot operation modes. The time-constraints imposed in this research are shown in Table 4.6 . These values can be reduced based on the application and can be determined by performing braking experiments, as described in [46, 85].

Table 4.5: Parameters used for the SSM safety configuration for different minimum distance calculation approaches.

TriSSM Approach		**Sensor Modalities**	
	Parameter	**ToF and Mocap-based**	**2D LiDAR-based**
TriSSM-Vo TriSSM-Vr	Time Intervals:	$T_{red} = 0.4s$,$T_{stop} = 0.4s$, $T_r = 0.1s$	
	Max. Directed Speeds:	$v_{r\,max} = 1.5m/s$,$v_{o\,max} = 1.6m/s$	
	Cushioning Constants:	$C_{dC} = 0.3m$, $C_{rec} = 0.17m$ $B_{min} = 0.2m$	$\lambda_{ows} = 0.82m$, $C_{dC} = 1.12m$
Tri-SM	Constant PSD(d_C) and RSD(d_R):	$\{0.5m, 1.1m\}$	$\{1.32m, 1.92m\}$

Table 4.6: Table showing the time-constraints ($T_{prev,curr}$) used in the Reflexxes Motion Library based online speed fraction controller for transitioning between normal ($\psi_{normal} = 2$), reduced ($\psi_{reduce} = 1$) and stop ($\psi_{stop} = 0$) robot operation modes.

Previous $\psi(t-\Delta t)$ \ Current $\psi(t)$	ψ_{normal}	ψ_{reduce}	ψ_{stop}
ψ_{normal}	-	**250ms**	**100ms**
ψ_{reduce}	**500ms**	-	**100ms**
ψ_{stop}	**1000ms**	**500ms**	-

The maximum magnitude of the end-effector velocity in the direction of motion (x and y) for the task in this experiment is $v_{TCP_{max}} = 1.5m/s$. The maximum speed a human agent or its body part moving has been defined in [85, 123] as $\approx 1.6m/s$. Here in this setup, the maximum directed speeds for **TriSSM-Vo** and **TriSSM-Vr** are chosen as $v_o max = 1.6m/s$ and $v_r max = 1.5m/s$, respectively. The braking distance $B_{min} = 0.2m$ for the robot end-effector is explicitly selected for this task. It was determined by repeating multiple braking scenarios when the robot reached the maximum speed during the task. The PSD and RSD for $TriSM$ safety approach can be determined by assuming no information is known for the human or robot speeds. For a 2D LiDAR, the human-robot minimum distance was measured from the circumference of the robot operating workspace. For this task, the radius of the operating workspace was, $\lambda_{ows} = 0.82m$.

Figure 4.25 shows the comparison of the minimum distance calculated with respect to the ground truth, d_{gt}. The RMSE of *Ideal* minimum distance d_{ideal} is approximately $96mm$,

which validates our assumptions of $d_{ideal} \approx d_{gt}$ for this experiment setup. The RMSE of the *Ideal* minimum distance from the ToF sensor arrays is 250*mm*, and *Lidar* is 358*mm*. The RMSE of *Lidar* is larger because it reports the distance from the fixed base of the robot and its visibility is limited to measuring the position of the human legs or the lower torso. The real ToF sensor arrays raw distances and the filtered (low-pass) distances are compared to the ground truth. A filtered distance results in a shift in the report of the sensor distances and attenuates the response. This is not desirable as SSM is time-constrained and can adversely affect the behavior and the reaction-time of the robot. Hence, filtering is not used during the implementation of the SSM. The uncertainty and spikes in the data in comparison to the emulated/simulated 8-node ToF sensor d_{Sim} are due to erroneous readings and blind spots. The RMSE of the ToF sensor is larger than the ideal sensor because of the sensor reading accuracy and the sparsity of the sensor nodes (eight nodes) that result in blind-spots and abrupt jumps in the range reading(s). This sensor detection uncertainty is compensated in the C_{dC} cushioning constant as Z_s, as described in Eq. 4.28.

Overall, it can be observed that on-robot ToF sensor arrays can be used for human-robot minimum separation distance in real-time. It is important to note that the ToF sensor nodes directly report the raw distance, and minimal computation is required for the minimum distance measurement. On the other hand, mesh-mesh distance calculations are done in the digital-twin between human-avatar and the simulated robot to determine the minimum separation distances which require extra computation [29, 76](as it searches for the closest point on the mesh of the human avatar to the mesh on the robot). Thus, with higher accuracy comes a computational cost. Thus, the limitations are hardware limitation of the sensor performance and the number of sensors in a ToF sensor array. A modular 32-node ToF sensor array prototype with a higher sampling rate is developed as part of this research. The details of the 32-node ToF sensor array are presented later in Chapter 6.

According to the safety-metric formulation in Eq. 4.55, the safest SSM configuration is the one where the robot does not move. But that also means there is no productivity. Hence, the '*usably safe*' SSM configuration that optimizes the safety and productivity should be preferred [85]. The graph shown in Figure 4.26 plots productivity against safety during a task performed with a given SSM safety configuration for all sensors. It is observed that the *Real* and *Ideal* based minimum distances have higher productivity than *Lidar*. TriSSM-Vr gives the highest productivity for *Real* and *Ideal*; however, this configuration is less safe.

	Ideal Sensor Model	Sim	Real	Lidar
RMSE(w.r.t. Ground Truth in meters)	0.096	0.156	0.251	0.358

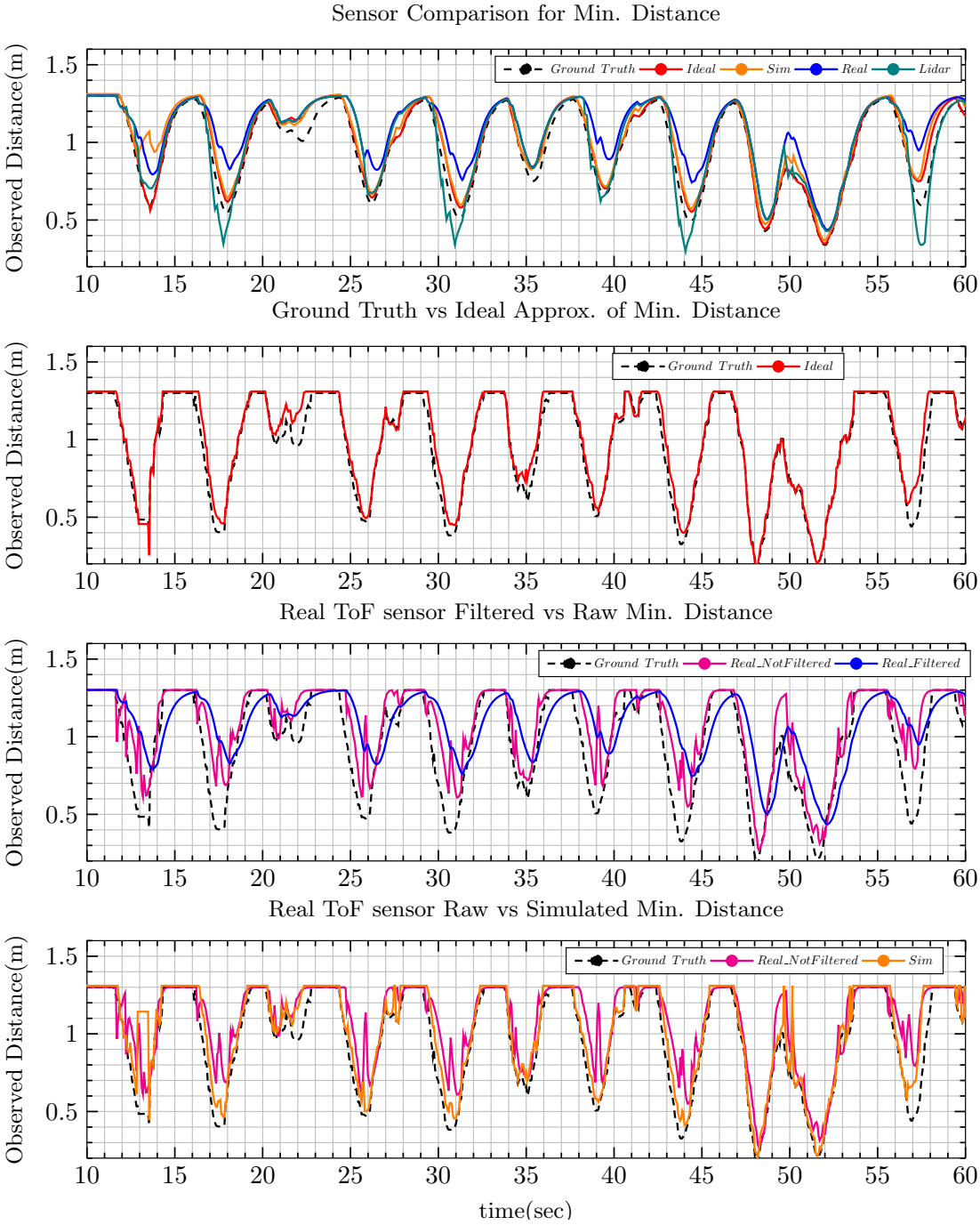


Figure 4.25: Minimum Distance Measurement Comparison of the implemented sensor modalities - *Real, Sim, Ideal & Lidar* with the ground truth in terms of RMSE.

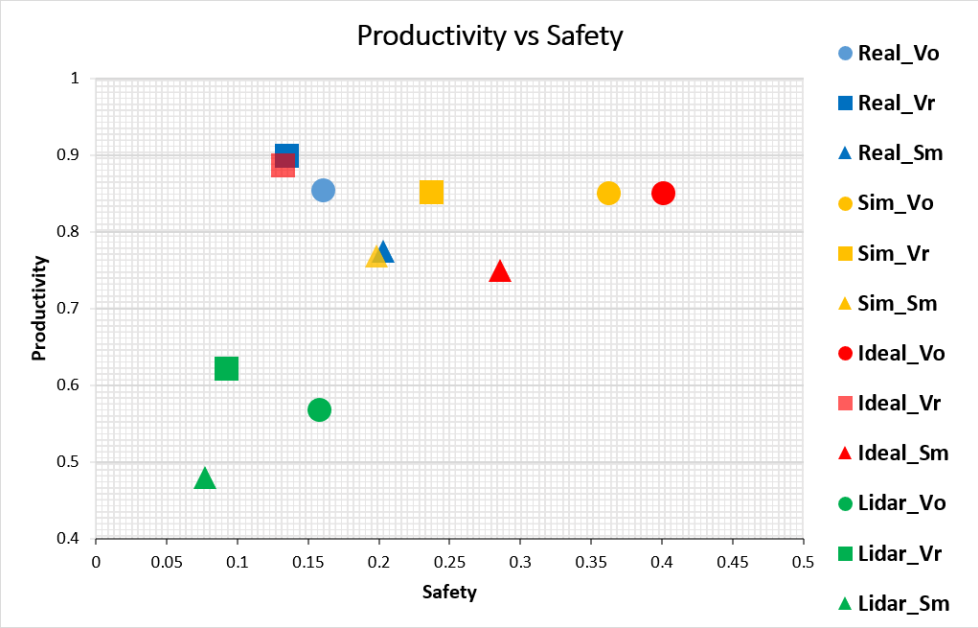


Figure 4.26: Productivity vs Safety Metric graph; for all SSM safety configurations implemented (*Vo,Vr,Sm*), for all minimum distance calculation approaches (*Real, Ideal & Lidar*).

The highest safety is reported for TriSSM-Vo for *Ideal* minimum distance. It can be further seen that for a given minimum distance calculation, TriSSM-Vo is safer than TriSM-Vr.

The *Lidar* based SSM reports lower safety levels. This is because a 2D Lidar needs to consider a larger intrusion distance human/obstacle ($C_{dC\,2D} = C_{dC\,ToF} + \lambda_{ows}$) compared to the smaller value considered for the Real and Ideal distances, refer Eq. 4.30 and Eq. 4.37 [46]. In order to keep the comparison same, for a given C, the *Lidar* based TriSSM safety configuration behaves relatively less safe in comparison to *Real* and *Ideal*. Increasing the intrusion distance makes the SSM configuration safer at the cost of decreased productivity. As the safety-metric is obtained from the ground truth and the robot TCP velocity V_{tool}, the *Real* sensors are less safe due to the error in minimum distance calculations. Thus, it can be observed that given the minimum distance calculation is accurate, a safer and more productive SSM configuration can be implemented by using on-robot sensors. Also, the consideration of robot and human/obstacle directed speeds in TriSSM-Vo adds to the safety and productivity. Using robot directed speed alone result in less safe SSM configuration.

For ease of visualization and comparison, the separation distances, velocity changes, reaction times, and robot's times to stop and reduce are plotted as radar graphs, as shown in

Figure 4.27. It can be observed in Figure 4.27(a) that the fastest reaction is of the TriSSM-Vo for the *Real* sensor setup. The reaction times are negative to denote that the robot anticipated a stop event before the minimum distance reached the critical distance threshold. The average reaction time of the system represents the sensitivity and responsiveness to distance and directed speed changes. The *Real* based SSM is more responsive as the time taken to determine the minimum distance is faster than the calculation of the *Ideal* distance.

The average stopping and reduce times indicate the anticipatory nature of human/obstacle motion in the shared workspace. As seen in Figure 4.27(b), the time to stop is higher for all Tri-SM, and the time to reduce speed is higher for TriSSM-Vr and Tri-SM. It can be observed that TriSSM-Vo gives the best results.

The 'velocity before stop' graph in Figure 4.27(c) represents how often there are sharp deceleration(s) at the stopping event. It can be seen that TriSSM-Vr has the biggest velocity change. This can cause more wear on the actuators, and sudden speed changes can be uncomfortable for the human sharing the workspace.

The separation distance gives an idea of when reduced speed or stop events were triggered and what the average separation distance. It is also indicative of the anticipatory nature of the SSM setup. The dynamic Tri-SSM based on *Real* and *Ideal* minimum distances have nominal stop and reduce distances averaged around $0.5m$ and $0.75m$. However, the *Lidar* is more conservative.

The Figure C.3 shown in Appendix C., shows the response of the robot TCP velocity based on the change in the measured minimum distance for all SSM and minimum distance configuration(s). The videos demonstrating the experiment setup and the performance of the ToF sensor arrays can be viewed at https://youtu.be/c9qDZBQ86qY and https://youtu.be/mQw0rl0upms, respectively. Next, we shed light on the limitations of the implemented dynamic SSM setup in the following section.

4.2.7 Limitations

The Tri-Modal dynamic SSM that leverages both the relative human-robot speeds and separation distance, results in a more consistent and smoother robot movement. The results of this work are intended to provide the design for a simple plug and play device as an

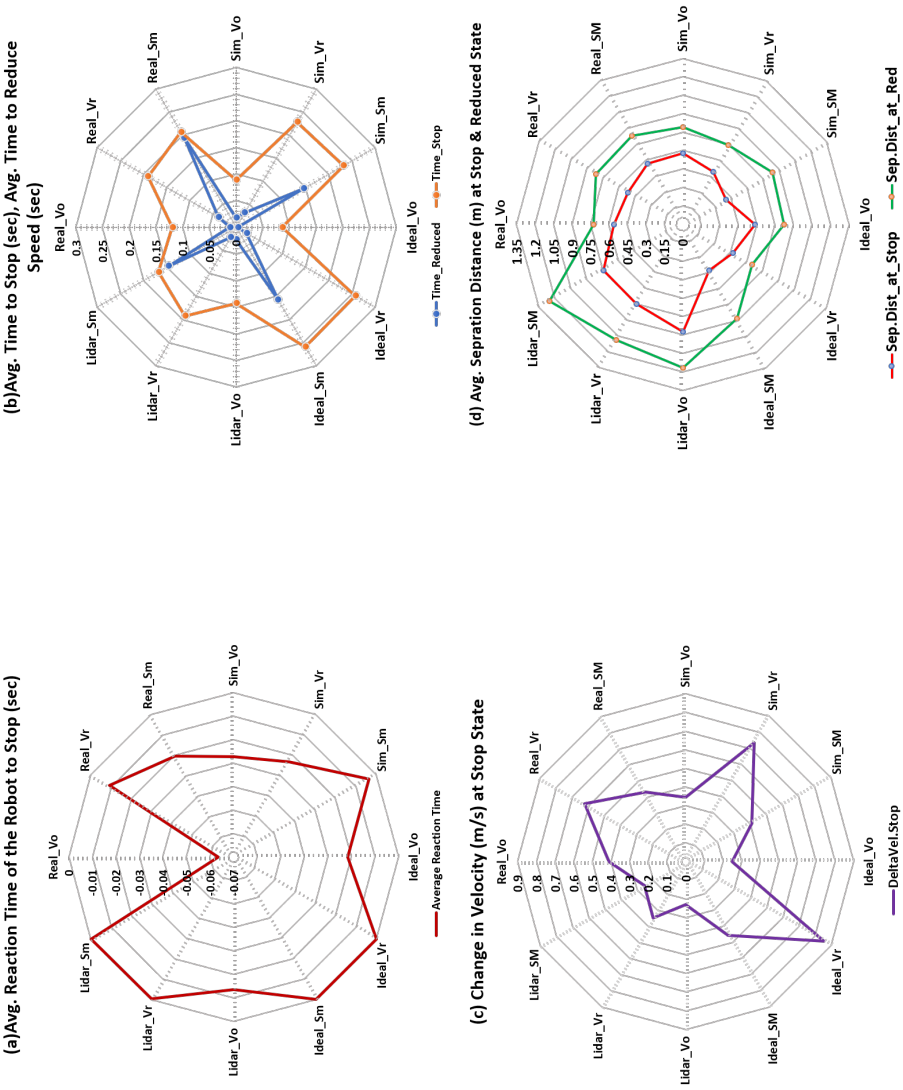


Figure 4.27: Radar graphs comparing a) Average Reaction Time of the system b) Average Time to Stop and Reduce c) The change in velocity at a Stop event d) Average human/obstacle robot separation distance at reduce and stop events; for all SSM safety configurations, for all minimum distance calculation approaches.

alternative or addition to current 2D LiDAR scanners for optimizing productivity while ensuring human safety. It was observed that if a minimum distance calculation is accurate, a safer and more productive SSM configuration can be implemented by using on-robot ToF sensors. Thus, the 16-node array prototype can be used for addressing the issues of sensor uncertainty due to a blind spot. A ToF sensor ring with 16 ToF sensor nodes is designed and tested, as described in Chapter 6.

The results of this implementation are for a human operator moving on a predefined path in the human-robot shared workspace. The human operator movement in industry while performing the task is not predefined and is unpredictable. Moreover, with a higher degree of collaboration, such as performing a shared pick-n-place task, the human subjective feedback in terms of comfort and measures of sharing the robot workspace along with the evaluation done here is needed. This experiment should be performed for more than one human operator.

In the next section, we present how the dynamic SSM setup can be extended to incorporate the mass and shape of the payload carried by the robot.

4.2.8 Extending the Dynamic SSM

In this research, we considered the human-robot separation distance and relative human-robot velocities for the formulation of SSM. The safety during SSM was achieved by ensuring that a collision does not occur by regulating the robot speed, thereby the overall robot energy, and maintaining a minimum distance from the human operator sharing the workspace. According to the ISO standards, the *post-collision control* methods such as Power and Force Limiting (PFL), consider the severity of the injury if the robot were to collide with a human. The risks are mitigated and identified by reducing the overall energy transferred at the moment of impact at an operational point on the robot end-effector/TCP.

For a fixed robot end-effector/TCP velocity, the robot carrying a payload at the end-effector will have higher energy and will pose more danger than when it is not carrying the payload. These changes in energy are dependent on the inertial properties of a robot manipulator at the end-effector can be represented by the effective mass of the robot. The effective mass formulation and derivation as detailed in [123, 124] is shown in Appendix A..

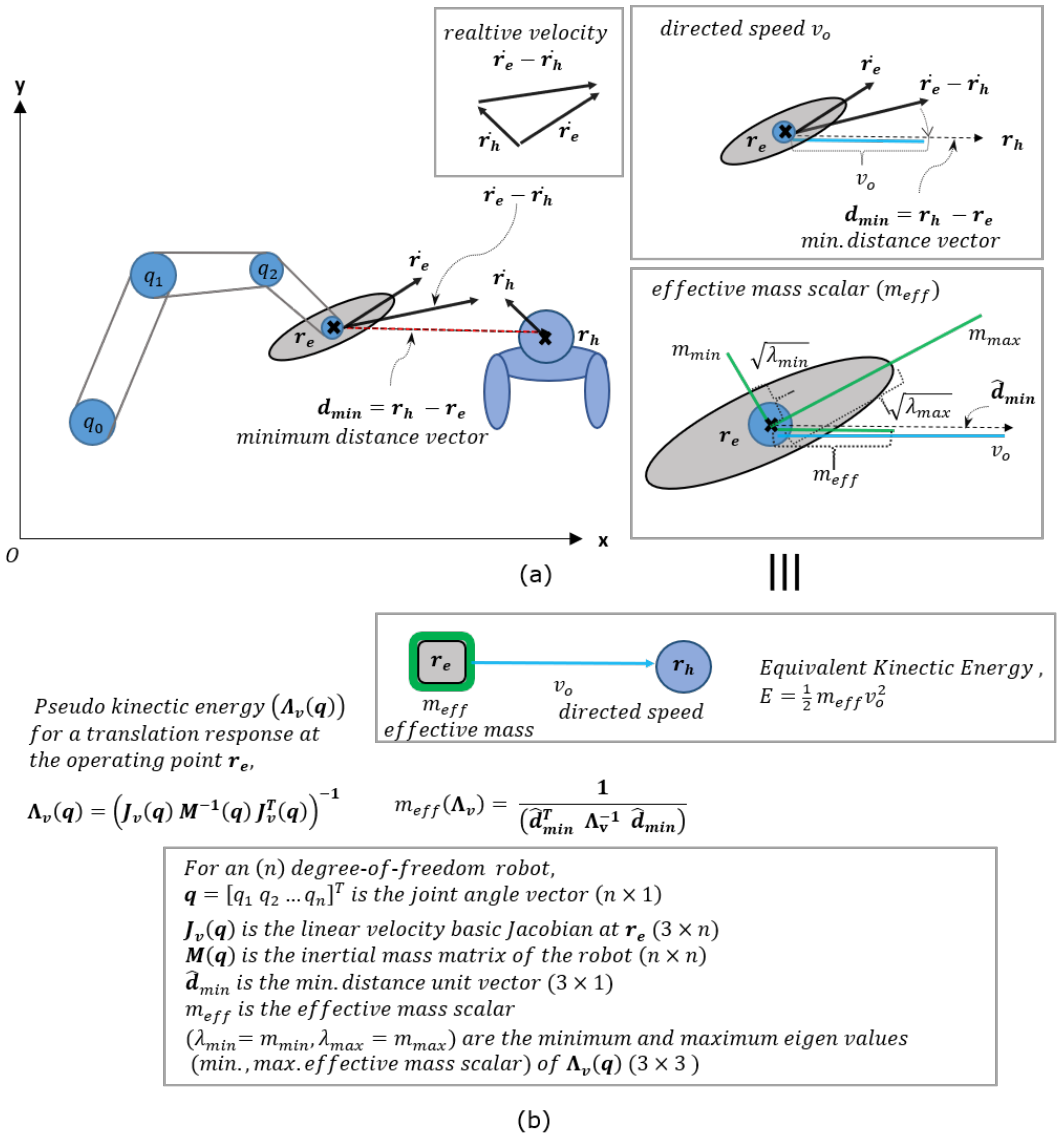


Figure 4.28: A planer example showing the formulation of an effective mass scalar m_{eff} at an operational point $\boldsymbol{r_e}$ on the robot end-effector moving at directed speed v_o towards the human.

In a dynamic SSM setup, the effective mass can be calculated in the direction of the human-robot minimum distance vector $\hat{\boldsymbol{d}}_{min}$ at an operational point ($\boldsymbol{r_e}$) on the end-effector moving at the directed speed v_o towards the human. A planer example showing the formulation is shown in Figure 4.28.

A robot manipulator carrying a payload and moving with an end-effector velocity $\boldsymbol{v_{TCP}}$ has a higher momentum and kinetic energy, thereby posing a higher danger in comparison

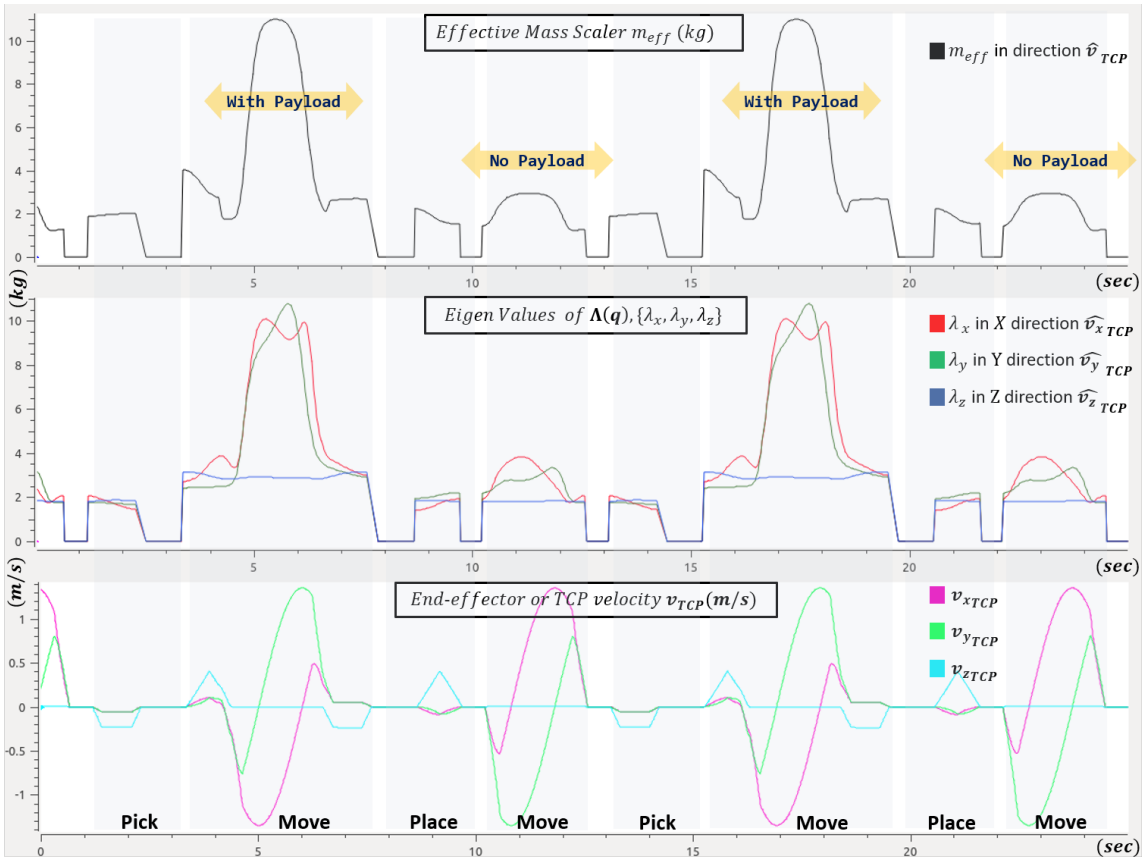


Figure 4.29: Results of the effective mass calculation for a standard pick and place task for a payload of $10kg$.(Top) shows the effective mass scalar m_{eff} in the direction of the end-effector/TCP velocity $\boldsymbol{v_{TCP}}$.(Middle) The eigenvalues $(\lambda_x, \lambda_y, \lambda_z)$ of the pseudo kinetic energy matrix Λ_q, which represent the effective mass scalar in the (x, y, z) directions.(Bottom) The components of the end-effector/TCP velocity $\boldsymbol{v_{TCP}}$.

to the robot carrying no-load moving at the same velocity. The results of the change in the effective mass with and without the payload calculated in the direction of the robot motion are shown in Figure 4.29.

Next, we will show how the change in the effective mass scalar with and without the payload can be accounted for in the calculation of protective/critical safety distance (PSD) in a dynamic SSM setup.

4.2.8.1 On accounting for the effective mass in Protective/Critical Safety Distance Calculation

In SSM the safety is achieved by reducing the overall robot energy by controlling the speed of the robot. The human-robot separation distance is compared to a protective/critical safety distance (PSD) defined as a threshold to determine the robot action to slow down or stop. The PSD $d_C(t_0)$ formulation as defined previously in Chapter 4 is defined as

$$d_C(t_0) \geq v_o(T_R + T_{stop}) + v_R T_R + B + C_{dC}$$

This is dependent on the human-robot directed speed v_o at a given time t_0. The magnitude of v_o directly relates to the distance from the possible point of impact on a human operator the that robot starts decelerating to stop at a distance equal to or greater than the cushioning constant C_{dC}. This can also be interpreted as the kinetic energy of the robot that needs to reduce to come to a complete stop. The kinetic energy (E_0) of the effective mass (m_0) representing a robot carrying no load at the end-effector and moving with a directed speed v_o towards the human can be defined as

$$E_0 = \frac{1}{2} m_0 v_o^2$$

The kinetic energy (E) for an effective mass (m_{eff}) for the robot carrying a payload at the same directed speed v_o would be greater than E_0. The relation between E and E_0 can be defined as

$$\begin{aligned} E &= \frac{1}{2} m_{eff} v_o^2 \\ &= \frac{1}{2} m_0 \left(\sqrt{\frac{m_{eff}}{m_0}} v_o \right)^2 \\ &= \frac{1}{2} m_0 \left(k_m v_o \right)^2 \; = \; k_m^2 E_0 \end{aligned}$$

where $k_m = \sqrt{\frac{m_{eff}}{m_0}}$ represents the scaling factor of the directed speed v_o. The above equation shows the relation that the kinetic energy E of the robot with payload (effective mass m_{eff}) moving at speed v_o is equivalent to a robot with no-payload (effective mass m_0) moving at

the scaled speed $k_m v_o$. This can be accounted for in the calculation of PSD as

$$d_C(t_0) \geq k_m v_o(T_R + T_{stop}) + v_R T_R + B + C_{dC}$$
$$where\ k_m = \sqrt{\frac{m_{eff}}{m_0}}$$

This would result in a more conservative behavior of the robot when carrying a payload than with no payload. The robot starts decelerating a greater distance from the possible point of impact on the human operator when carrying a payload than with no payload.

In addition to the mass, the shape of the payload/object can also pose a danger to the human-operator during the HRC task. In the next section, we detail how the shape of the payload/object can be accounted for in the calculation of the PSD.

4.2.8.2 On accounting for the shape of the payload/object in Protective/Critical Safety Distance Calculation

The danger posed by a robot carrying a sharp object towards the human poses a greater danger than a blunt object. At the point of impact/collision between a human and a robot the severity of injury due to the sharpness of the object would be higher compared to the collision with a blunt object. This can be characterized by defining the energy density (ED) at the point of impact (r_e) of the object. The energy density (ED) is the ratio of the kinetic energy of the effective mass (m_{eff}) moving in the direction of impact and the surface area of the payload/object in contact with the human-operator (a_e). The biomechanical criteria relating the energy density to the severity of the injury have been researched in [79, 86] and detailed in the ISO standards [43].

The sharpness of the object can be defined using the surface area vector. A surface area vector has the magnitude of the area of the surface and direction of the vector normal to the surface where the surface area is calculated. A few examples for a cuboid, a cylinder, and a sharp object are shown in Figure 4.30(a). The $\boldsymbol{a_{max}}$ vector of the objects represents the vector normal to the blunt side, which has the largest surface area and poses the least danger at impact. Conversely, the $\boldsymbol{a_{min}}$ vector is normal to the sharpest side or edge of the object. The $\boldsymbol{a_{side}}$ is another vector normal to other sides of the object that have a surface area such that $\|a_{min}\| \leq \|a_{side}\| \leq \|a_{max}\|$.

For safe human-robot interaction, an ideal scenario is when the blunt side of the payload/object ($\boldsymbol{a_{max}}$)is always aligned with the direction of the robot end-effector towards the human-operator (minimum distance vector $\boldsymbol{d_{min}}$), which can be represented by the cross-product $\boldsymbol{a_{max}} \times \boldsymbol{d_{min}} = 0$. Conversely, the sharpest side of the object ($\boldsymbol{a_{min}}$) should be facing in the opposite direction or is orthogonal to the minimum distance vector, which can be represented by the dot-product $\boldsymbol{a_{min}}^T \boldsymbol{d_{min}} = 0$. The surface area a_e is calculated at the operational point $\boldsymbol{r_e}$, which is the point of possible impact with human-operator at $\boldsymbol{r_h}$, such that $d_{min} = \|\boldsymbol{r_h} - \boldsymbol{r_e}\|_2$. It is challenging to calculate the surface area $a_e = \|\boldsymbol{a_e}\|$ as it changes with the robot motion and the minimum distance vector $\boldsymbol{d_{min}}$. Hence, we estimate a_e as the component of the resultant surface area vector of the blunt and sharpest area vector, $a_e = \hat{\boldsymbol{d_{min}}}^T (a_{min} + a_{max})$.

The energy density ED_0 for the safest scenario for a robot end-effector moving with a directed speed of v_o towards the human-operator can be defined for the largest surface area a_{max} and the smallest effective mass m_0 as

$$ED_0 = \frac{E_0}{a_{max}} = \frac{m_0 v_o^2}{2a_{max}}$$

The energy density (ED) for an effective mass (m_{eff}) for the robot carrying a payload at the same directed speed (v_o) with the surface area a_e at the possible point of impact towards the human can be defined as

$$ED = \frac{E}{a_e} = \frac{m_{eff} v_o^2}{2a_e}$$

The energy density (ED) with a heavier effective mass ($m_{eff} > m_0$) and smaller surface area ($a_e < a_{max}$) would be greater than ED_0. The relation between ED and ED_0 can be defined as

$$\begin{aligned} ED &= \frac{m_{eff} v_o^2}{2a_e} \\ &= \frac{m_0}{2a_{max}} \left(\sqrt{\frac{m_{eff}}{m_0}} \sqrt{\frac{a_{max}}{a_e}} v_o \right)^2 \\ &= \frac{m_0}{2a_{max}} (k_m k_a \, v_o)^2 \;=\; (k_m k_a)^2 ED_0 \end{aligned}$$

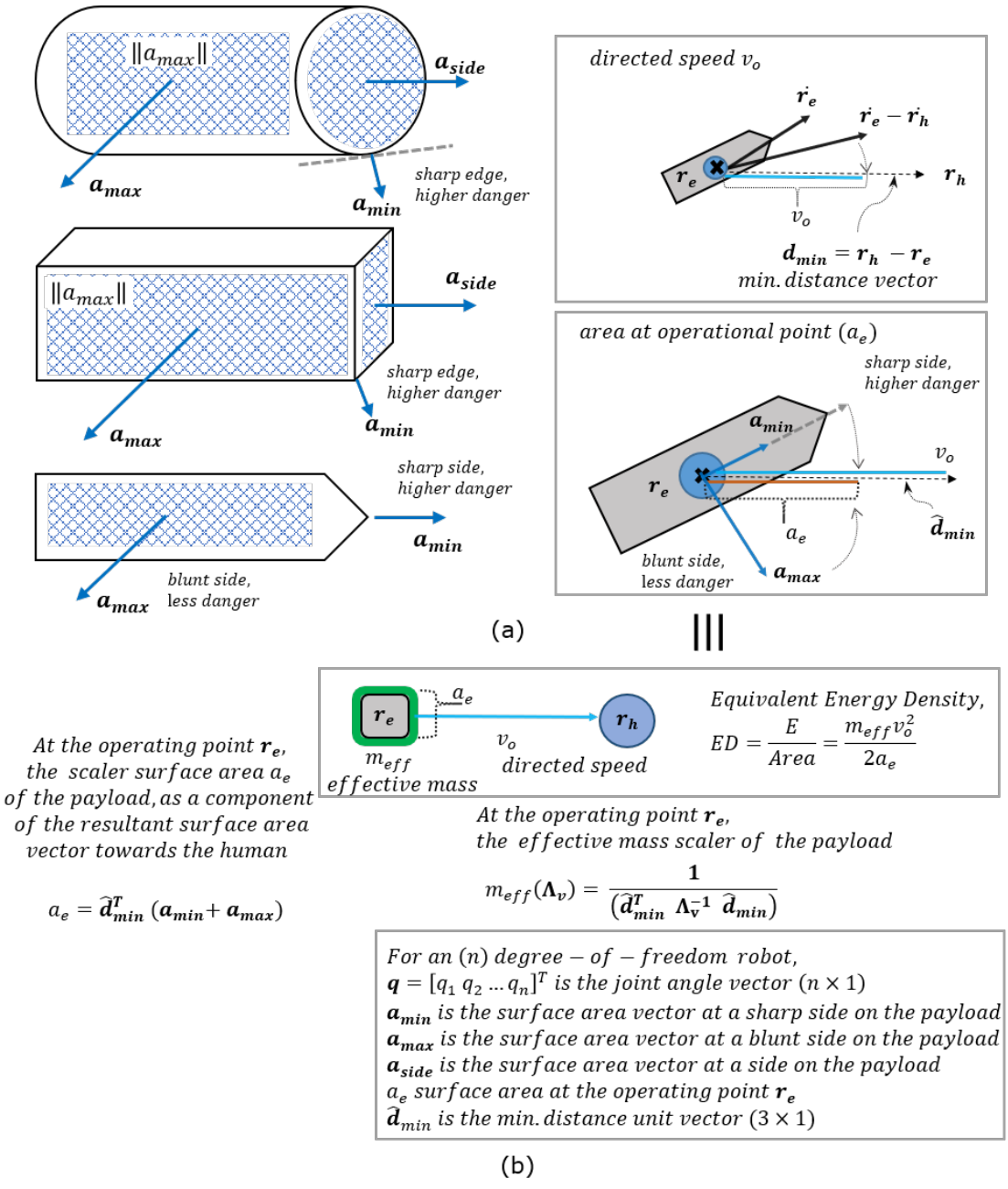


Figure 4.30: A planer example showing the formulation of the surface area a_e at an operational point r_e on the robot end-effector moving at directed speed v_o towards the human.

where k_a and k_m represent the scaling factors of the directed speed v_o proportional to the decreasing surface area a_e and increasing effective mass m_{eff}. The above equation shows the relation that the energy density ED of the robot with payload (effective mass m_{eff}) and the surface area a_e moving at speed v_o is equivalent to a robot with no-payload (effective mass m_0) and the maximum surface area a_{max} moving at the scaled speed $k_m k_a\, v_o$. This can be accounted for in the calculation of PSD as

$$d_C(t_0) \geq k_m k_a v_o (T_R + T_{stop}) + v_R T_R + B + C_{dC}$$
$$where\ k_m = \sqrt{\frac{m_{eff}}{m_0}}\ \ and\ \ k_a = \sqrt{\frac{a_{max}}{a_e}}$$

In the next section, we summarize the results of the validation and implementation of the dynamic SSM setup and conclude.

4.2.9 Intermediate Conclusion

A dynamic Speed and Separating monitoring setup using on-robot Time-of-Flight ranging sensor arrays is successfully implemented based on the SSM formulation as outlined in the research [85] and ISO standards for robotic safety. From the simulation and real world implementation results, it can be concluded that there is a significant advantage to using on-robot ranging sensors for SSM in comparison to the 2D LiDAR. The proposed TriSSM-Vo/Vr safety configurations with on-robot ranging sensors are over 40% more productive than 2D LiDAR-based Tri-SM configurations currently used in industry. Listed below are the main conclusions and observations:

- **On Human-Robot Minimum Distance Accuracy**: The minimum distance calculation based on the ellipsoidal approximation of robot links is validated. The human-robot minimum distance measured from the center of the ToF sensor array rings placed at the center of the robot links can be used as an approximation to the absolute/ground truth minimum distance. This low-complexity direct computation methodology of minimum separation distance using on-robot ToF sensor arrays was successfully used in the implementation of dynamic Tri-Modal SSM safety configuration for an HRC setup.
- **On Safety and Productivity**:

  - Real and Ideal based SSM have higher productivity in comparison to 2D LiDAR-based SSM.
  - TriSSM-Vo is safer than TriSSM-Vr safety configuration for all sensor modalities. TriSSM-Vo based on Ideal measurements was observed to be the safest configuration.
  - TriSSM-Vr is more productive than TriSSM-Vo for all sensor modalities.
  - Tri-SM is the least productive for all sensor modalities.
  - The most optimal SSM safety configuration in terms of safety and productivity was TriSSM-Vo with Ideal sensor distance measurements.
  - A better accuracy in the minimum distance measurement increases the safety of an SSM configuration.
- **On Performance**:
  - In Real ToF-based SSM safety configurations, the robot behavior is more responsive to the changes in minimum distance and directed speeds. This responsiveness is due to the faster calculation of the minimum distance in comparison to digital-twin simulation and motion capture-based Ideal measurements. TriSSM-Vo based on real ToF-sensors is the most sensitive and responsive.
  - For TriSSM-Vo based SSM safety configurations, the robot behavior is more anticipatory than TriSSM-Vr and Tri-SM.
  - In TriSSM-Vr based SSM safety configurations, the robot motion has a sharper deceleration before stop, resulting in higher jerks and sudden speed changes.
  - For SSM safety configuration using 2D LiDAR, the robot behavior is conservative, and as a result slower.

Next, unlike the human walking on a fixed path, we use this setup with human-subjects while performing a defined assembly task. The subjective feedback using questionnaires and interviews is quantified to validate the conclusions and claims derived in this chapter.

Chapter 5

Results of Human Subject Experiments for Dynamic SSM

The implementation of dynamic SSM safety configuration using on-robot ToF laser ranging sensor arrays placed on the robot was achieved successfully. It was observed that in terms of safety and productivity, TriSSM-Vo safety configurations were optimally balanced. The overall performance of SSM safety configurations using *Real* and *Ideal* sensor modalities performed better in comparison to 2D *LiDAR*. These observations were made for experiment with fixed human-robot interaction as the human subject walked a predefined path in the shared workspace.

In this chapter, the dynamic SSM safety configurations are evaluated for an HRC setup where the human agent performs an assembly task with the robot. The experiments are performed with 19 human subjects with all the SSM safety configurations described in Chapter 4 [1]. The results are evaluated based on the objective criteria of safety, performance and productivity as mentioned in Section 4.2.5. In addition, subjective responses from the human subjects are collected using questionnaires and interviews following each experiment. The human subjects are asked to rate the behavior of the robot for a given SSM safety configurations in terms of their preference to work with, the overall speed, predictability, and comfort level.

The objective of these human-subject experiments is to validate and corroborate mainly the following three claims from the experiments in the previous Chapter 4:

(i) The least productive dynamic TriSSM approaches using the *Real* (8-node ToF sensor array prototype) and the *Ideal*(motion capture and digital-twin) sensor modalities are more productive than all the Tri-SSM approaches using 2D *LiDAR* sensor.

(ii) A dynamic TriSSM safety approach that considers the relative velocity of both human-agent/object (non-stationary) and the robot links (TriSSM-Vo) is safer, more anticipatory

[1] This human subject research was approved by the IRB, and the research statement of the experiments is presented in Appendix D.)

(predictable) and comfortable for the human agent to work with, than the safety approach that considers just the velocities of the robot links (TriSSM-Vr).

(iii) The uncertainty and error in minimum distance measurements using an 8-node ToF sensor array prototype (*Real*) compared to the motion capture (*Ideal*) for a dynamic TriSSM safety configuration affects the perceived safety and comfort level of the human-agent during the task.

In addition to the dynamic SSM safety, the human physiological signals are also recorded while performing the experiments for future use. A subsystem is integrated with the dynamic SSM setup to monitor and record the human physiological signals and the human-robot collaboration state information synchronously and concurrently. The experiment setup is described in the following section.

5.1 Experiment Setup

The experiment setup and workspace are similar to those described in the previous chapter. The collaborative work-space and the robot motion parameters are not changed. The HRC setup consists of individual tasks for the robot and the human. The HRC experiment setup is shown in Figure 5.1.

The human task during the experiment is to assemble a product consisting of three parts, as shown in Figure 5.2(a). One of the three parts, *Part-2*, is provided to the human by the robot. The other parts are assumed to be available in bins with known locations. The human picks the three component parts of the assembly from bins on tables marked as *Table-0*, *Table-1*, and *Table-3*, and the assembled part is placed in the bin on *Table-4* in the shared work-space, as shown in Figure 5.1 and Figure 5.2(b). The robot picks *Part-2* from a pallet consisting of 10 parts from *Table-2* and places in the bin on *Table-1*.

During the pick and place operation, the robot operating workspace overlaps with the human motion during the assembly, as shown in Figure 5.3. The safety measure during this setup are the dynamic SSM safety configurations previously described. To reiterate, the SSM safety configurations; TriSSM-Vo, TriSSM-Vr, and **TriSM**, are implemented using sensor modalities *Ideal* (digital-twin and motion capture), *Sim* (simulated 8 Node ToF sensor arrays and motion capture), *Real* (8-Node ToF Sensor array prototype), and 2D *LiDAR* (a simulated 2D LiDAR and motion capture).

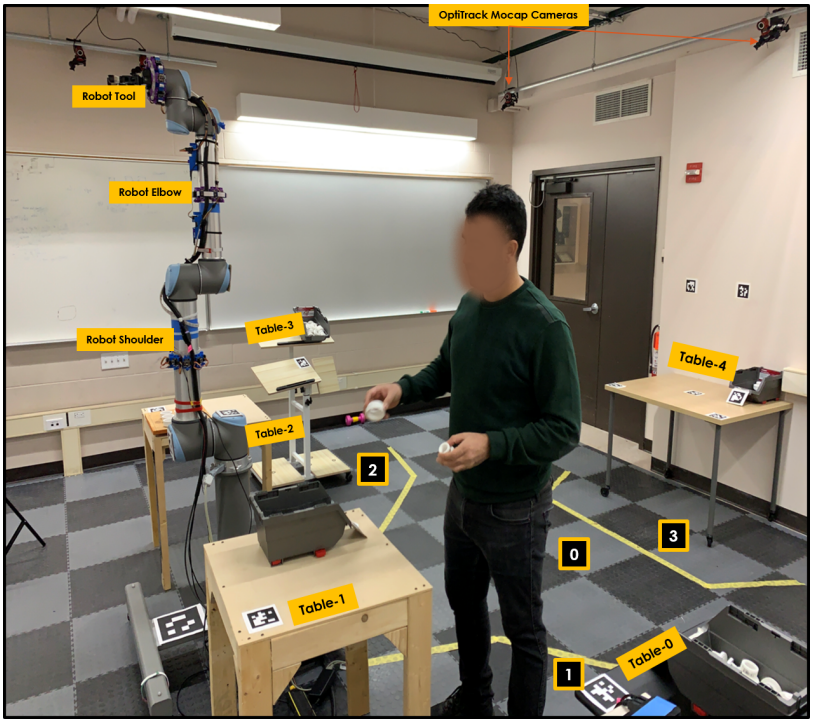


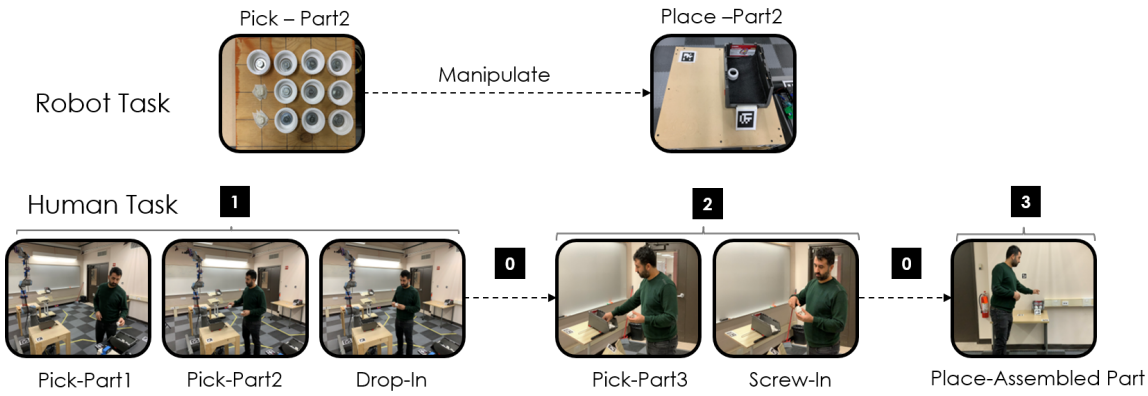
Figure 5.1: The HRC experiment setup showing the human performing the assembly task. During this task the robot picks parts from a pallet on Table-2 and places it on the bin on Table-1.

Based on the robot operating workspace and the human position in the shared workspace, safety zones are demarcated, as shown in Figure 5.3(b). These safety zones in terms of the human pose are termed as *safe*, *warning* and *danger* (refer Section 3.1.4.1 and [111]). It should be noted that these safety zones are *static* in nature and are used for auto-generation of event markers using the human position in the zones during the recording of the data.

The task progress is tracked based on the human position in the shared workspace. A top view of the human subject's task flow is shown in Figure 5.3(a). The task steps for human subject for assembling the part are described below:

- **Human Task Step, s1**: Pick *Part-1* and *Part-2* from *Table-0* and *Table-1*, respectively. In this step, the human subject creates a sub-assembly by mating the parts. This action is referred as a 'drop-in', because the mating is achieved by dropping *Part-1* in a through hole slot in *Part-2* (shown in Figure 5.2(a)).
- **Human Task Step, s2**: The human subject picks the *Part-3* from *Table-3* and completes the assembly by 'screwing it' to the sub-assembly. This action is referred as a 'screw-in'.

(a)



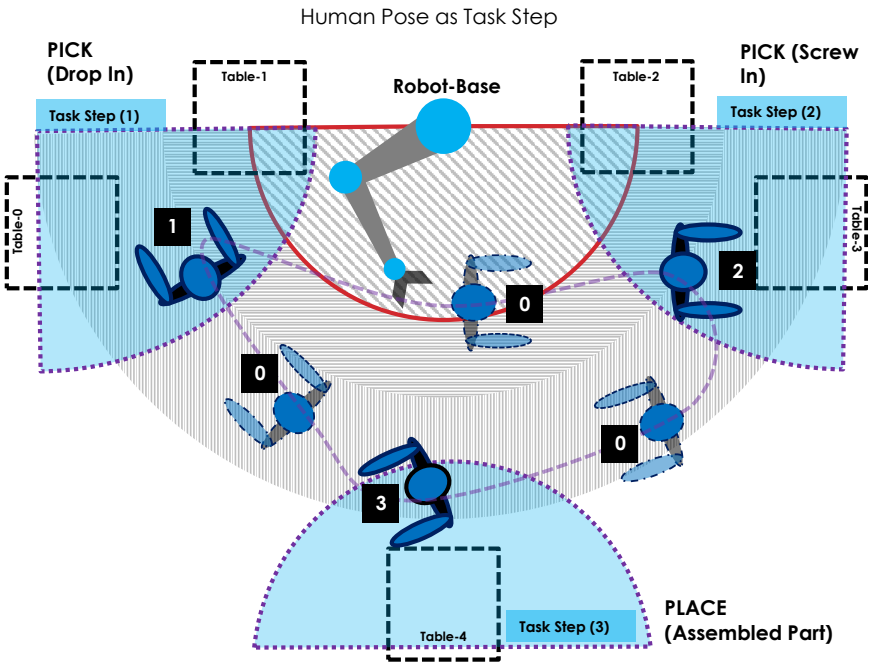
(b)

Figure 5.2: (a) Steps for assembly of the part for human during the task. (b) Human and robot task steps during HRC experiment. The robot picks Part-2 from the pallet and places it in a bin. The task steps performed by the human for completing the assembly.

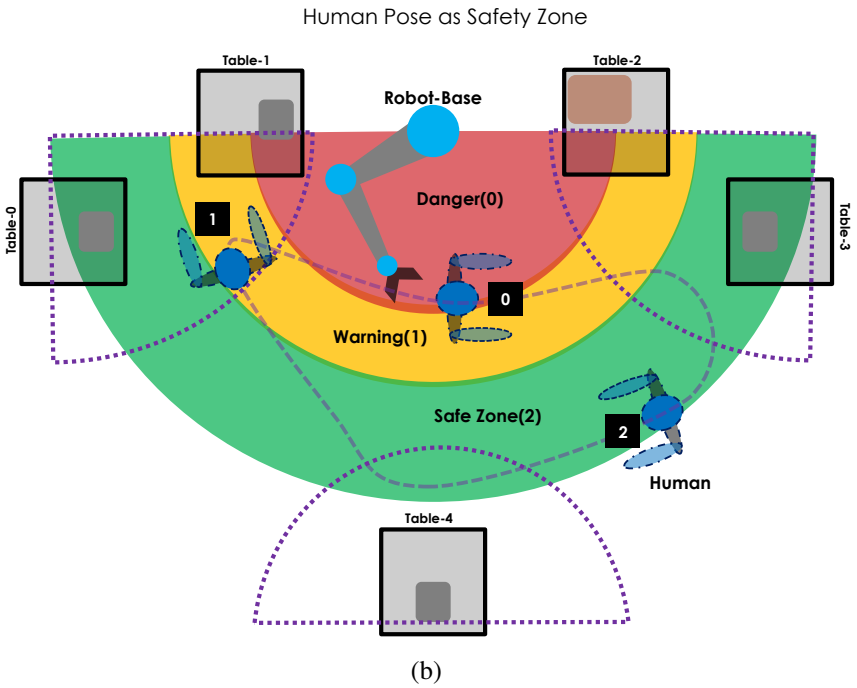
- **Human Task Step, s3**: The human subject places the assembled part in a bin on *Table-4*.
- **Human Task Step, s0**: This task is the human movement between the tables during the assembly.

Therefore, at any given moment, a human position during the HRC experiment can be defined by a *safety zone* and a *task step*. For example, in Figure 5.1, the human pose is identified as ***'warning:s1'*** .

In conventional setups of SSM, LED indicators have been used as a form of visual feedback indicating the safety stop (red), reduced speed (yellow/orange), and normal (green) robot operation modes [13, 83, 111]. An LED indicator is placed at the base of the robot, as shown in Figure 5.4. In addition to the LED indicator, RGB LED light strips are coiled around the robot links signaling the same color scheme. During the human subject experiment, gaze tracking, as well as human subject feedback, is used to determine if the LED indicators and RGB light strips are helpful to the subject as visual feedback during the



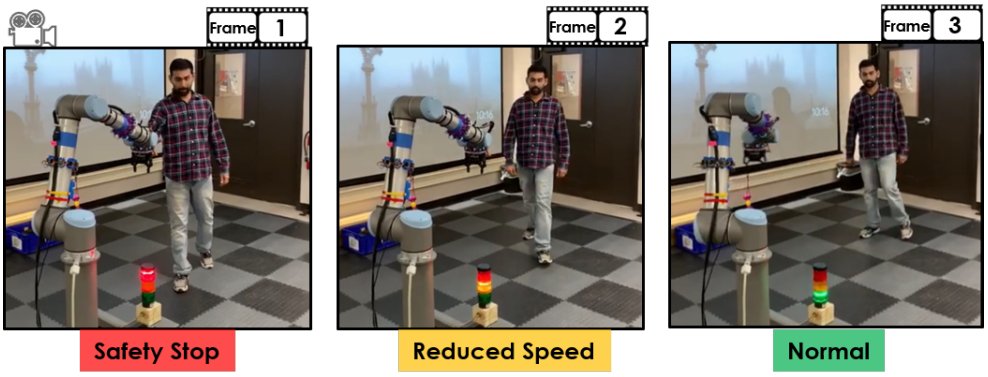
(a)



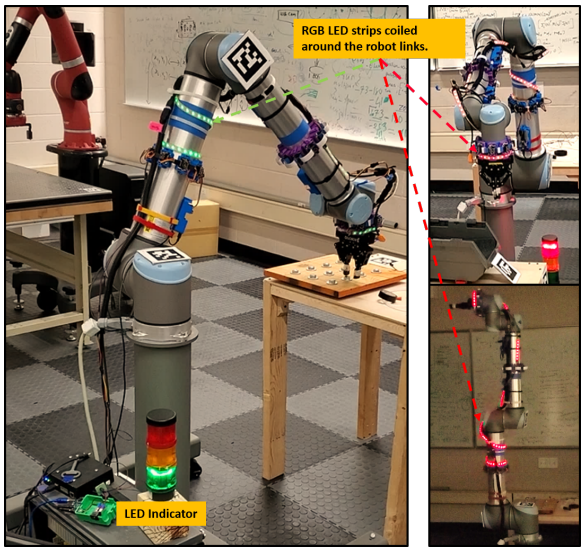
(b)

Figure 5.3: A birds-eye view of the collaborative workspace depicting (a) Human Pose as Task Step, and (b) Human Pose as Safety Zone.

experiment. The use of RGB LED strips as a form of signaling devices industry has been researched in [100].



(a)



(b)

Figure 5.4: (a) LED indicators as visual feedback representing the robot operation mode, where Red-Safety Stop,Yellow-Reduced Speed,Green-Normal Video Link (b) A flexible RGB LED light strip coiled around the robot links. The visual feedback schema is the same as the LED Indicator.

Next, we describe the human-subject experiment procedure in the following section.

5.1.1 Experiment Procedure

During the experiment, the human subject performs the assembly task with the robot performing pick and place operation for one of the components needed for assembly. Before the experiment begins, the human waits near *Table-4*, where the human pose is noted as ***safe:s3*** and the robot is in the home position (as shown in Figure 5.1). The experiment is started when the robot begins its motion. It is stopped when the robot returns to its home position after picking the ten parts (*Part-2*) from the pallet and placing them in the bin on *Table-1*. The combined task is started when the robot moves into a picking position over the pallet on *Table-2*, and the human begins moving to pick up the parts for assembly. The task is considered complete when the robot places the last (the 10th *Part-2*) into the bin for assembly.

Before the start of each experiment, the human subject has an extra four *Part-2*s in the bin on *Table-2*, and the robot supplies the rest. Thus, the maximum number of assemblies by the human during the experiment is fourteen. This ensures that the human is able to continue assembling and not wait for the robot to supply the parts at the start of the experiment. The human subject continues the assembly till the end of the experiment. The other component parts, *Part-1* and *Part-3*, are always available in the bins for assembly.

In a single session with a human subject, a total of nine experiments for a span of *45-60* minutes are performed for a subset of the twelve Tri-SSM safety configurations. The first two experiments are baseline experiments. The first baseline experiment is the human performing the assembly task with no robot movement (as previously referred to in Section 4.1.3.1 to as *NoHRI*). In this case, the human is given ten parts (*Part-2*) beforehand in the bin, and is asked to complete ten assemblies. To mitigate a learning bias for assembly, the human subjects is asked to practice assembly in the workspace to familiarize him/herself with the task. The time taken to complete the baseline task is used to calculate productivity loss when the robot is in motion for an experiment with a given SSM safety configuration.

For the human-subject to have a robot behavior as a reference to compare with other experiments, the second baseline experiment is done with the robot. For each experiment, an SSM safety configuration is used to ensure safety during the HRC task. During the experiment, the human-subject is asked to notice changes in the robot behavior, if any. The human subject does not have prior information about which safety configurations is used

Table 5.1: This table shows the subject groups A,B and C , and the different SSM safety configurations used for the robot behavior during the HRC experiment.

Subject Group	Sensor Modality	Tri-SSM Safety Config.			Experiment Group Objective
		Vo	Vr	SM	
A	Real	X		X	Compare the performance of 2D LiDAR-based safety configurations conventionally used in industry to the least productive Real and Ideal Tri-SM configurations.
	Ideal	X		X	
	LiDAR	X	X	X	
	Sim				
B	Real	X	X		Compare the performance of SSM safety configuration of TriSSM-Vo and TriSSM-Vr for Real and Ideal sensor modalities.
	Ideal	X	X	X	
	LiDAR				
	Sim	X	X		
C	Real	X	X	X	Compare the performance of SSM safety configuration of TriSSM-Vo , TriSSM-Vr and Tri-SM for the Real sensors.
	Ideal	X		X	
	LiDAR				
	Sim	X	X		

in an experiment.

There are in total twelve different safety configurations, three safety algorithms TriSSM-Vo, TriSSM-Vr, and Tri-SM implemented using measurements from sensors *Real*, *Ideal*, *LiDAR* and *Sim*. As mentioned previously in Chapter 4, the *Ideal* and *Sim* measurements are from the same sensor modalities; the digital-twin simulation and motion capture. Hence, *Sim* is not considered for comparison for sensor modalities but is used to compare the performance of the algorithms.

The human subjects are divided into three groups with different combinations of SSM safety configurations. They are shown in Table 5.1.

As observed previously, the 2D LiDAR-based SSM was the least productive and conservative in comparison to other sensor modalities (refer Section 4.2.6). Therefore, the subject group A is chosen to compare the performance of *2D LiDAR* based safety configuration with the *Real* and *Ideal*-based **TriSM** configuration. The highlighted rows and columns in the table represent the main experiments for a subject group. This experiment group's objective is to compare the performance of LiDAR-based SSM safety configurations to the least productive Real and Ideal based Tri-SM safety configurations.

The experiments in Group B are done for comparison of safety algorithms TriSSM-Vo and TriSSM-Vr for *Real* and *Ideal* sensor modalities. The experiments in Group C are chosen for comparison of all the safety algorithms based on Real sensors. In this group, a comparison to that of the *Ideal* sensors is also made. The experiments for each subject group are performed in random order. For all the subject groups, the safety configuration

for the second baseline experiment with the robot is chosen randomly from the safety configurations of a sensor modality not being used during the experiment. For subject group A, the choice for the baseline is from {Sim-Vo, Sim-Vr, Sim-SM}, for B and C is from {LiDAR-Vo, LiDAR-Vr, LiDAR-SM}. This is done to ensure that each subject experiences the robot behavior based on all sensor modalities and different SSM safety algorithms. Thus, an assumption is made that the responses for all SSM safety configurations from human subjects are independent.

Next, the we present the details of the cyber-physical system (CPS) implemented for performing HRC experiments.

5.1.2 A Cyber-Physical System (CPS) for HRC Experiments

A Cyber-Physical System (CPS) for HRC is implemented by integrating the existing dynamic SSM setup, and the subsystem for concurrent collection of human-robot state information and human-physiological signals. The existing dynamic SSM setup, as described in the previous Chapter 4 in Section 4.2.5, provides the human-robot state information such as human pose, robot operation mode, robot state information (Robot TCP velocity, joint position and speed fraction), human-robot minimum separation distance, and human-robot directed speeds. Using this information, event markers are generated (refer Appendix B.), and the human physiological data is synchronized with the human-robot collaboration information (detailed in the previous work [3]). In order to visualize this data, a plotting tool is integrated, and both the human bio-metrics and the human-robot state information is plotted. In addition, an external camera records the video of the human performing the experiment. The video frames are also synchronized and collected with the system. This can be used by the investigator during post processing to replay the event recorded. Thus, this can be used for analysis tool for a complete human-in-the-loop system. A frame for the video of the interface is shown in Figure 5.5.

In the following section, the approach for using human gaze tracking and fixation to determine human attention during the experiment is discussed.

Figure 5.5: A Frame of Video showing the complete system implemented, with robot and human bio-metrics plotted and recorded in real-time. Video Link

5.1.3 Real-Time Identification of Objects using Gaze Tracking

The gaze tracking data can be used to measure the human fixation and attention on different objects in the shared human-robot workspace. The Pupil-Lab headset has a camera representing the human vision perspective. This camera is called the 'gaze tracking camera' or 'the world-view camera.' The Pupil-Lab eyeglasses track the pupil movement and find the corresponding 2D point (u, v) in the image frame of the gaze tracking camera, as shown in Figure 5.6. Prior to the experiment, the gaze of the human subject is calibrated to the pupil movement.

In order to identify objects in the 2D image, there are three approaches. They are listed as follows:

- Record the video of the camera and use image processing to identify the objects. This can be done in real-time or after post-processing. However, as the camera is moving, the image in the video would be riddled with motion artifacts. The object identification would require training of the objects in the workspace. A poorly trained detection algorithm could result in low accuracy of detection of objects. As this approach would require the generation of a training data-set of images of objects in the work-space, this option was not deemed feasible.

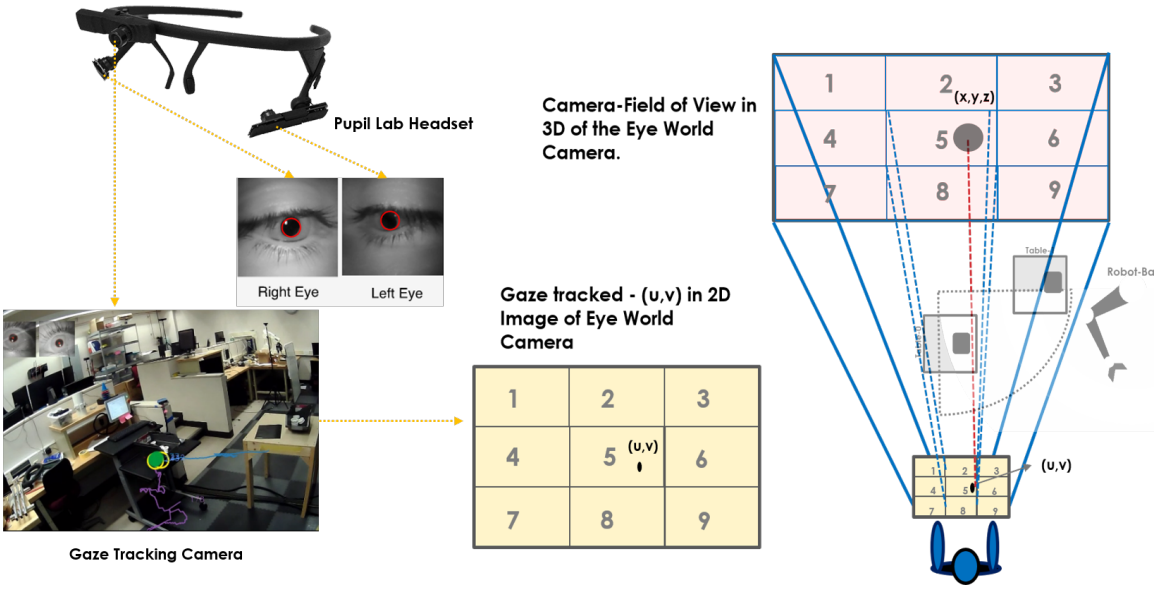


Figure 5.6: A representation of the gaze tracking image coordinates from the Pupil World Camera used to determine a 3x3 grid, projected into the 3D space. The objects in the FOV of the gaze-tracking camera are correlated to the objects in the FOV 3D grid of the simulated camera in the digital-twin setup.

- Placing identifiable April-Tag or ArUCO markers, used for localization of moving camera reference frame with respect to the markers [125]. These markers are commonly used in augmented reality applications. Each marker can be identified in an image with a unique identification tag or number. By placing markers on the objects of interest in the shared workspace, the frame of the video can be post-processed to identify the location of the markers in the 2D image frame, and compared it to the tracked gaze point (u, v). In this experiment, April-Tag markers have been placed throughout the workspace. This methodology can provide very accurate identification of objects seen by the human using the tracked human gaze information. However, this method suffers from motion artifacts and markers are not always visible in each image frame. To track the objects in real-time, it adds more complexity to the system.
- This approach uses the digital-twin to identify objects in the field of view (FOV) of the human. In this experiment setup, the human position and the head orientation information are available in real-time using the motion capture system. The Pupil-Lab headset is stationary relative to the head and therefore it can be tracked. In the digital-twin setup,

a human avatar mimicking the motions of the human subject is available. Using this information a 'camera twin' of the gaze tracking camera in the simulation is implemented. The FOV of the camera is set as 100 deg diagonal as specified by the Pupil-Labs. The 2D gaze information is then classified in 3x3 grids in 2D, and projected in the 3D space of the 'camera twin'. As the entire workspace is simulated in the digital-twin, the objects of interest in the FOV of the camera twin can be identified and reported in real-time. This approach, although approximate, gives real-time identification of objects paid attention to by the human subject. This setup is shown in Figure 5.6. For more accurate gaze tracking, the number of grids can be increased (for example, 5x5 or 9x9 grids of the FOV of the 'camera twin').

The third approach of using a simulated 'camera-twin' is implemented and integrated in the digital-twin used for the dynamic SSM setup. During the experiment, the objects of interest tracked in the shared-workspace are the tables (*Table-0, Table-1, Table-2, Table-3, and Table-4*), LED indicator at the base of the robot, and the robot links. The robot links tracked are the tool (TCP/end-effector), elbow, and shoulder link, where the LED RGB strips are coiled and also the ToF sensor arrays placed. A frame of the video showing the implementation is shown in Figure 5.7.

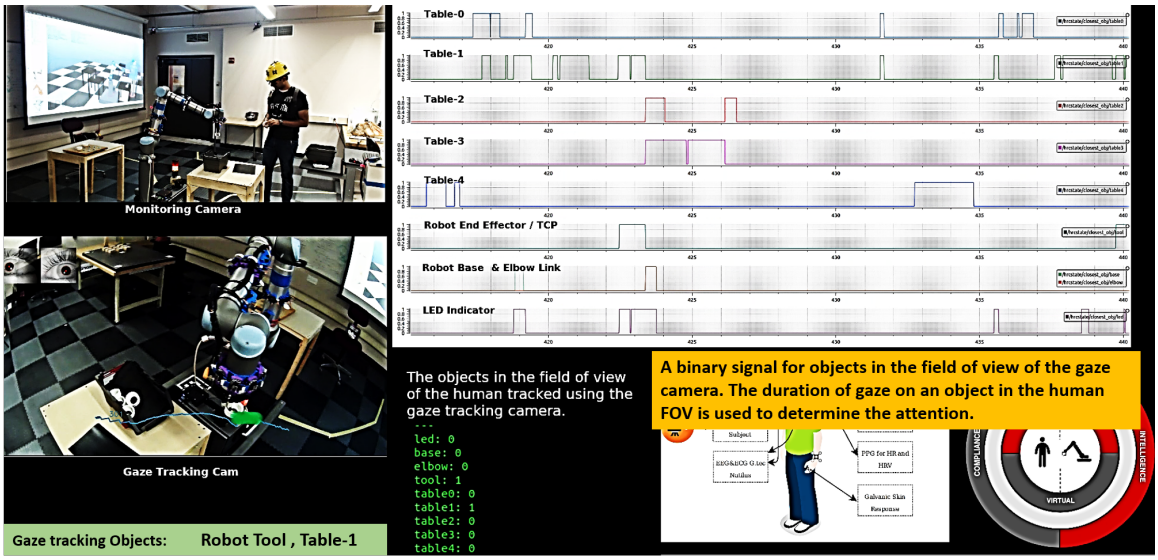


Figure 5.7: A frame of video showing the live gaze tracking of objects in the human-robot workspace. Video URL: https://youtu.be/6Nd11szwwgI?t=96

Next, we present the human-subject experiment results for different dynamic SSM configurations.

5.2 Experiment Results

In this section, the experiment is objectively evaluated in terms of safety, performance, and productivity and its result presented and discussed. The criteria for evaluation are the same as mentioned previously in Chapter 4, Section 4.2.5. In addition, subjective evaluation of the safety configurations is also performed following each experiment. The human subject is asked to fill a questionnaire and rate the robot behavior on a Likert Scale (1-Worst to 5-Best) in terms of preference to work with, overall perceived speed of the robot, safety/comfort, and anticipatory/intuitive nature of the robot during the task. Observations from the investigator about the number of parts assembled by the human subject at the end of each experiment and also instances when the robot lags behind to supply the parts for assembly are also tabulated and observations made. The results of the gaze tracking showing the attention of the human subject while performing the tasks are presented and discussed.

Next, we look at the demographics of the 19 human subjects who evaluated the robot behavior for different SSM safety configurations while performing the experiment.

5.2.1 Human Subject Summary

The results presented in this section are for experiments conducted with 19 human subjects. The demographics of the subjects for the HRC experiment in terms of gender, age, and experience or familiarity working with types of machinery such as CNC, lathe, or robots, is shown in Table 5.2. The results of the experiment from all the subjects are then sorted based on the SSM safety configurations. For brevity, the SSM configurations are referred from here on in this form: *Sensor Modality- Safety Algorithm*. For example, a SSM safety configuration using TriSSM-Vo and Real sensors is named as Real-Vo. The SSM safety configurations are sorted as Ideal-SM, Ideal-Vo, Ideal-Vr, LiDAR-SM, LiDAR-Vo, LiDAR-Vr, Real-SM, Real-Vo, Real-Vr, Sim-Vo, Sim-Vr, and Sim-SM.

In the next section, we present the objective evaluation of the dynamic SSM safety configurations as measured using the HRC state information from the digital-twin, motion

Table 5.2: The demographics of the subjects for the HRC experiment in terms of gender, age, and experience, or familiarity working with machineries such as CNC, lathe, or robots.

Subjects Gender	**Count**	**Avg. Age**	**Avg. rating in terms of experience and familiarity working with machineries such as CNC, lathe or robots, on a Likert Scale - (1-Less Familiar to 5-very familiar)**
Males	16	20.5	2.875
Females	3	23.3333	2.3333
Summary	**Total**= 19	**Avg.**= 20.9473	**Avg.**= 2.7894
	18 Right Handed, 1 Left Handed	Subjects in Group A =7, Subjects in Group B =6 Subjects in Group C =6	

capture, ToF sensor arrays, and the UR10 robot.

5.2.2 Objective Evaluation

The evaluation of the experiments for all dynamic SSM safety configurations is done in terms of safety and productivity. The average *Safety Metric* (Eq. 4.55) and the average *Productivity* (Eq. 4.56) for all subjects for a given SSM safety configurations are calculated. The time taken for the robot to pick and place ten parts without a human is measured as $t_{NoHRI} = 130secs$, which is constant for all experiments. For each experiment, the productivity is calculated according to Eq. 4.56. The average productivity is measured as the mean of the productivity for all the experiments of a dynamic SSM safety configuration. Similarly, the least safe moment during an experiment, which is the safety metric as defined in Eq. 4.55 is calculated. At the moment, the ground truth minimum distance between the human and the robot, and the velocity of the robot tool/end-effector, are also recorded. These values are averaged for experiments with the same dynamic SSM safety configuration. The Productivity vs. Safety Metric graph is shown in Figure 5.8(Top) for all the SSM safety configurations.

It is observed that the *Real* and *Ideal*-based minimum distance have higher productivity than *Lidar*. However, the highest productivity is given by TriSSM-Vo and TriSSM-Vr for *Real* ToF sensor array measurements. These are over 24% and 22% more productive in comparison to LiDAR-SM, which is conventionally used in industry. However, this configuration is less safe in comparison to TriSSM-Vo for *Ideal* and *Sim* sensors. The highest safety is reported for TriSSM-Vo for *Ideal* minimum distance. It can be further seen that for a given minimum distance calculation, TriSSM-Vo is safer than TriSSM-Vr.

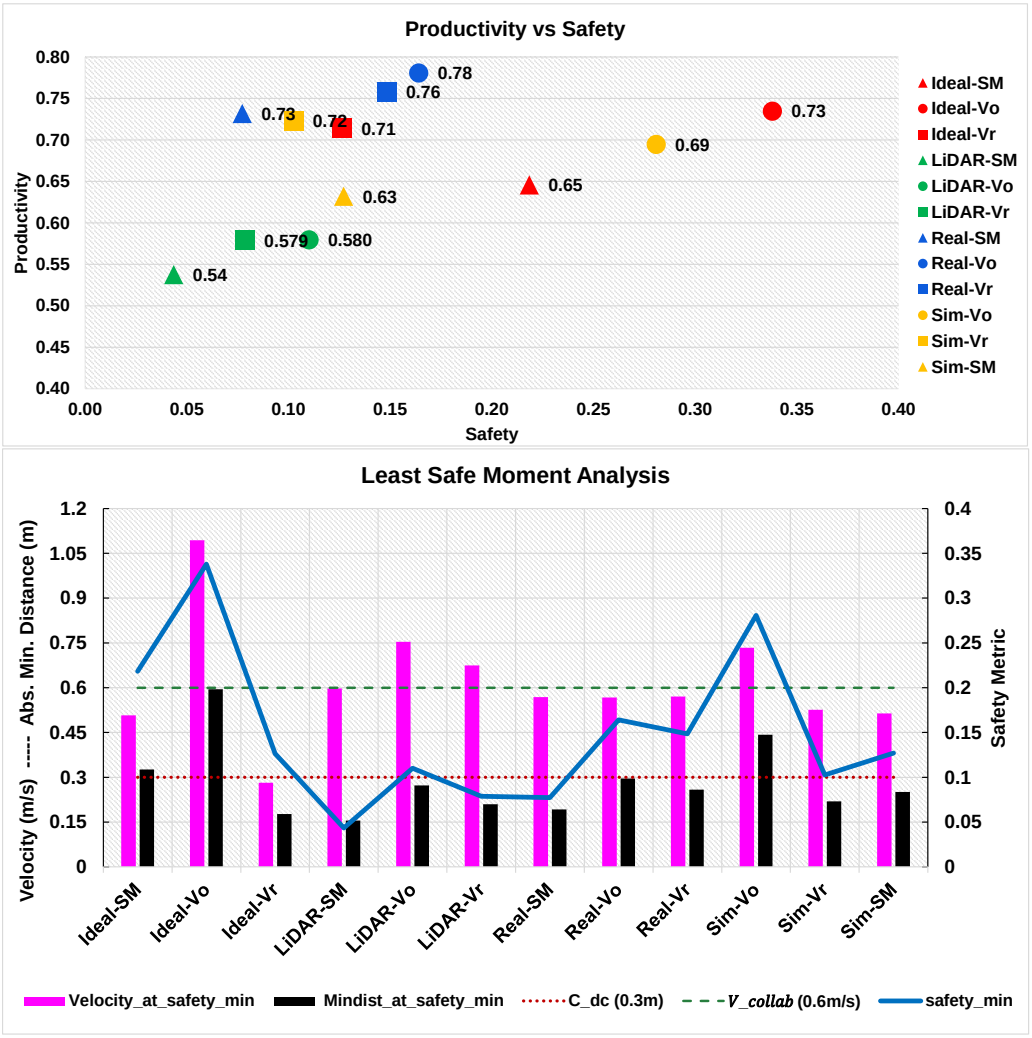


Figure 5.8: (Top) A Productivity vs. Safety Metric Graph; for all safety configurations. (Bottom) Analysis using the safety-metric, representing the average velocity and average minimum distance measured at the least safe moment during the experiment.

The safety of the SSM configuration are also analyzed by comparing the average absolute minimum distance and average velocity of the end effector at the least safe moment (safety metric) during the experiment. The results are shown in Figure 5.8 (Bottom).

According to the research presented in [43, 51, 123], for a robot of the size of UR10 and not carrying a heavy load (or a sharp object) moving with a tool velocity, $v_{collab} \equiv V_{TCP} < 0.6m/s$, can be considered in a collaborative mode, and poses negligible risk for the human due to a collision. The robot moves slow enough for the human to feel comfortable avoiding it. On the other hand, the minimum cushioning constant distance (C_{dc}) as defined earlier, is $0.3m$ (refer Section 4.2.6). This is the distance traversed by the robot tool to come to a complete stop when moving at maximum allowable speed. The safety metric is directly proportional to the square of the minimum distance and inversely proportional to the velocity of the end effector. Hence, the undesired safety scenario is when at the least safe moment during the experiment, the velocity of the tool is greater than $\leq V_{collab}$, and the absolute minimum distance is less than $\geq C_{dc}$.

From the graph shown in 5.8 (Bottom), it is observed that the Ideal-Vo and Sim-Vo are safer than the rest because the absolute minimum distance (ground truth, d_{gt}) at the least safe moment is greater. This is observed for all TriSSM-Vo safety configuration for a given sensor modality. The 2D LiDAR reports lower safety because the absolute minimum distance at the least safe point is small. This is the case because of the error in minimum separation distance measurements with a 2D LiDAR in comparison to the absolute/ground truth human-robot minimum separation distance. It measures the distance only in 2D and doesn't consider the reach of human arms. This becomes more apparent when the human subject is performing a task. The performance of the SSM safety setup is also evaluated. The average number of stop and reduced events, the average separation distance at a stop and reduced mode state, and the change in velocity at a stop state, are plotted as radar graphs, as shown in Figure 5.9.

It is observed from Figure 5.9(a) that TriSSM-Vo based safety configurations have a higher number of stop and reduce events across all sensor modalities. Amongst all, the Real-Vo has the highest number of stop and reduced events. This is because the *Real* sensors with a TriSSM-Vo safety setup, are more responsive and sensitive to changes in human-robot minimum distances and directed speeds. This observation coincides with the previous observation made in Section 4.2.6.

	Stop Events			Reduced Events			Avg. SepDist at Stop(m)			Avg. SepDist at Reduced Speed(m)			Change in Velocity(m/s) at Stop State		
Sensor-Algo	AVG	STD-S	STD-ERR	AVG	STD-S	STD-ERR	AVG	STD-S	STD-ERR	AVG	STD-S	STD-ERR	AVG	STD-S	STD-ERR
Ideal-SM	14.692	4.423	1.227	33.769	5.644	1.565	0.453	0.015	0.004	0.768	0.039	0.011	0.432	0.077	0.021
Ideal-Vo	27.286	8.222	2.197	62.357	15.280	4.084	0.610	0.040	0.011	0.746	0.041	0.011	0.423	0.129	0.034
Ideal-Vr	18.200	5.404	2.417	31.200	3.962	1.772	0.478	0.024	0.011	0.608	0.050	0.022	0.568	0.156	0.070
LiDAR-SM	28.000	3.536	1.581	41.000	4.743	2.121	0.601	0.105	0.047	0.939	0.051	0.023	0.245	0.042	0.019
LiDAR-Vo	43.500	17.647	6.239	83.125	20.918	7.396	0.729	0.060	0.021	0.909	0.071	0.025	0.190	0.058	0.021
LiDAR-Vr	35.714	12.352	4.669	50.857	8.030	3.035	0.688	0.068	0.026	0.840	0.098	0.037	0.304	0.060	0.023
Real-SM	16.222	7.102	2.367	43.222	7.311	2.437	0.516	0.166	0.055	0.811	0.089	0.030	0.380	0.149	0.050
Real-Vo	75.769	19.720	5.469	116.000	23.854	6.616	0.657	0.052	0.014	0.703	0.050	0.014	0.269	0.060	0.017
Real-Vr	40.250	7.815	2.763	66.375	10.141	3.585	0.614	0.077	0.027	0.711	0.059	0.021	0.487	0.112	0.039
Sim-Vo	32.667	8.742	2.524	68.917	14.563	4.204	0.570	0.042	0.012	0.720	0.078	0.022	0.402	0.139	0.040
Sim-Vr	25.000	8.089	2.860	47.250	10.566	3.736	0.552	0.036	0.013	0.764	0.036	0.013	0.494	0.113	0.040
Sim-SM	13.000	0.000	0.000	35.500	2.121	1.500	0.401	0.050	0.036	0.757	0.009	0.006	0.441	0.011	0.008

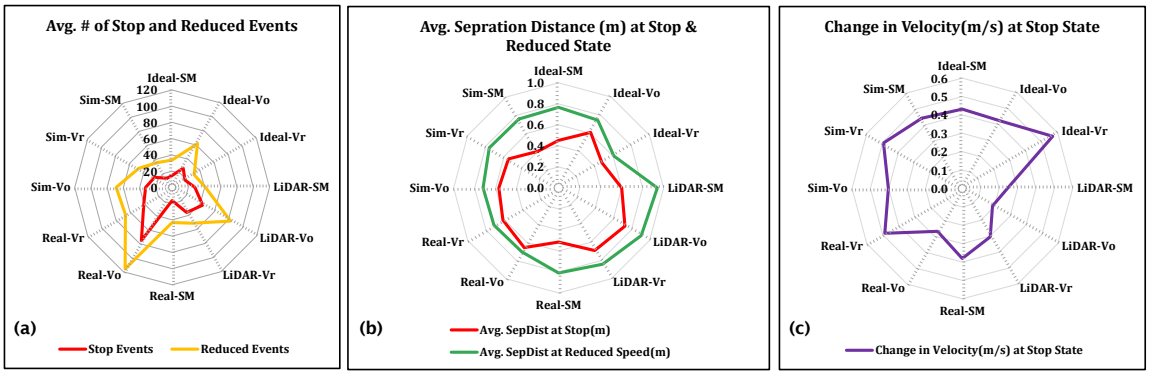


Figure 5.9: Radar graphs representing the performance analysis during the experiments comparing (a) number of safety stop and reduced events, (b) average human-robot separation distance at stop and reduced events, (c) average change in velocity at stop state; for all SSM safety configurations.

The average separation distances at stop and reduced state for all safety configurations are plotted, as shown in Figure 5.9(b). This data is indicative of when reduced or stop events were triggered, and what was the average separation distance. The results are similar to previously observed in Chapter 4. As expected, the SSM based on *Real* and *Ideal* minimum distances have nominal stop and reduce distances averaged at $0.53m$ and $0.74m$, respectively. However, the *Lidar* is more conservative, and the stop and reduce distances are averaged at $0.67m$ and $0.89m$, respectively. This observation is indicative that the overall speed of LiDAR is slower as the robot is stopped or moving slowly for a longer duration during the task.

The change in velocity at the stop state is defined as the difference in the end-effector velocity before and after the stop event has triggered. The average change in velocity is plotted and shown in Figure 5.9(c). This is indicative of higher speed changes and sudden movements. It is expected that this would cause human discomfort during the experiment. The highest change in velocity is measured for TriSSM-Vr for all sensor modalities, with

Ideal-Vr having the sharpest changes. This result coincides with earlier observations.

To summarize, Real-Vo is the most productive, and Ideal-Vo is the safest dynamic SSM safety configuration. The objective observations indicate that Ideal-Vr would be the least comfortable to work with. The LiDAR-based SSM configurations are the slowest and least productive. Next, we look at the results of the subjective evaluation done for 19 human subjects who evaluated the robot behavior for different SSM safety configurations while performing the experiments.

5.2.3 Subjective Evaluation

The subjective evaluation following each experiment is performed in three steps. First, following the experiment, the human-subject is asked to fill a questionnaire shown in Table 5.3. Second, the human subject is asked to rate the experiments performed during a session on a Likert scale (a scale of *1 to 5*, with *5* being the highest). The rating is in terms of best preference to work with, overall perceived speed of the robot, their level of safety/comfort, and predictable or anticipatory nature of the robot. The third is the observations of the total number of parts assembled at the end of the experiment by the human subject. The principal investigator keeps track of the efficiency of the robot to deliver the part (*Part-2*) to the human for assembly.

5.2.3.1 Questionnaire

The list of questions on the questionnaire asked to the human subject following each experiment for an SSM safety configuration,and its significance are shown in Table 5.3. The results are averaged across all subjects for a given SSM safety configurations. The resulting plots are shown in Figure 5.10. The Likert scale of *1 to 5* is interpreted as 1-very less, 2-less, 3-neutral, 4-more, and 5-most.

For the response for *Q1*, it is observed that on an average all human subjects felt calm working with the robot for all safety configurations. The human subjects felt calmer with Real-Vo and Ideal-Vo SSM configurations. The human-subjects did experience surprise when working with Ideal-Vr safety configuration. The human subjects on an average did not feel anxious working with the robot. However, they did feel more anxious with the Ideal-Vr SSM safety configuration.

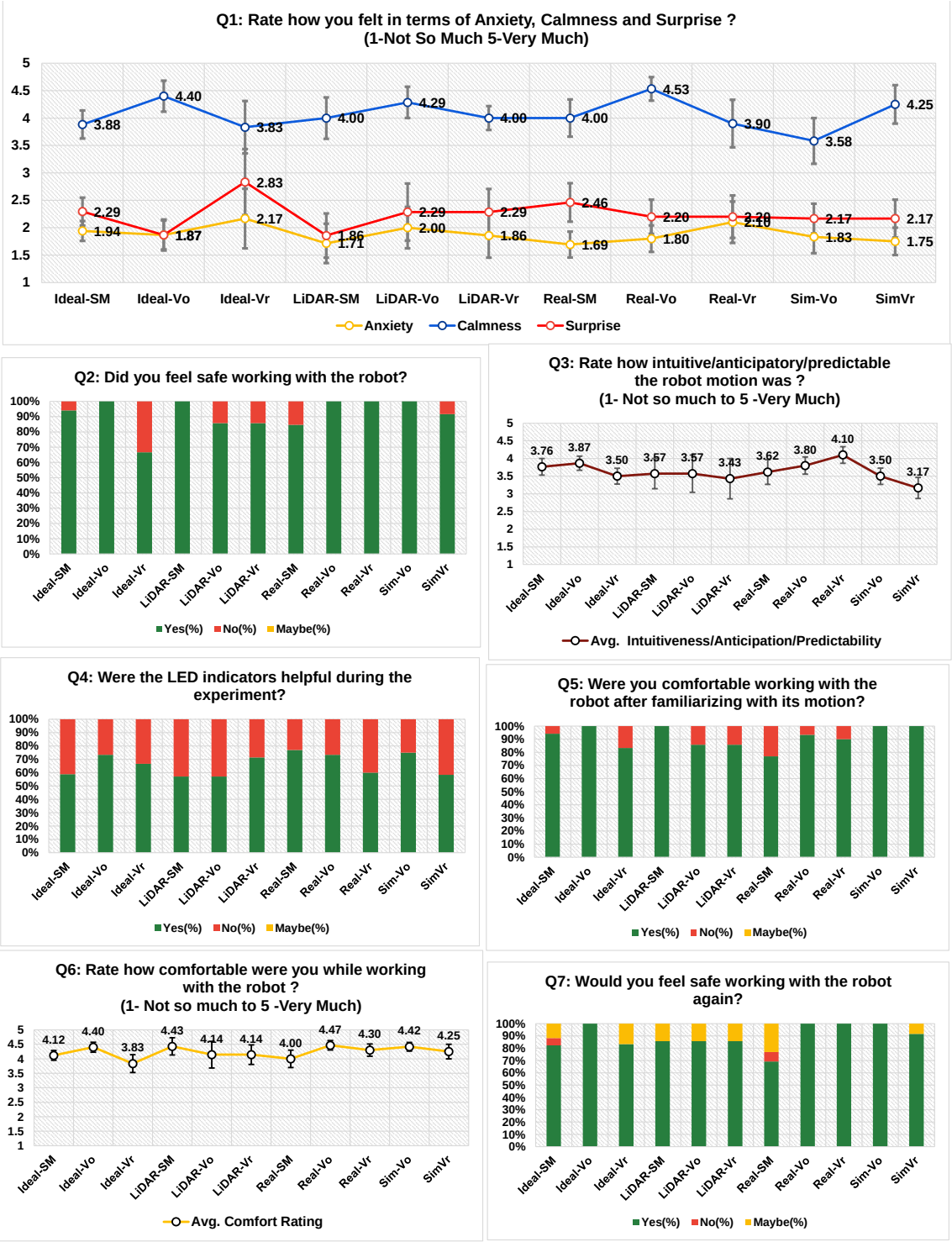


Figure 5.10: Graphs showing the mean analysis of the responses for the Questionnaire filled by the human subject after each experiment.

Table 5.3: A list of questions asked to the human subject following an experiment for an SSM safety configuration.

	Question		Response Type	Scoring	Significance
Q1	Rate how you felt during the experiment in terms of	Anxiety –	Likert Scale 1 to 5	–	The response provides insight into the mental state of the human during the experiment. It is desired that the human is less anxious, not surprised, and calm during human-robot collaboration.
		Calmness –	Likert Scale 1 to 5	+	
		Surprise –	Likert Scale 1 to 5	–	
Q2	Did you feel safe working with the robot? –		Yes, No, Maybe	+	A general inquiry regarding the human subjects sense of safety during the experiment.
Q3	Rate how anticipatory/intuitive/predictable the robot motion was?		Likert Scale 1 to 5	+	Was the robot behavior within expectations of the human subject.
Q4	Were the LED Indicators helpful during the experiment?		Yes, No, Did Not Notice	+	To see whether visual feedback is helpful to the human subject during the experiment.
Q5	Were you comfortable working with the robot after familiarizing with its motion? –		Yes, No, Maybe	+	To see if the human subject feels more comfortable with the robot behavior during the experiment.
Q6	Rate how comfortable were you while working with the robot?		Likert Scale 1 to 5	+	A rating on the human subjects comfort level sharing the robot work-space.
Q7	Would you feel safe working with the robot again?		Yes, No, Maybe	+	To see if the human subject would prefer the robot behavior to do more assembly tasks.

The response to *Q2* showed that 33% of the human subjects did not feel safe working with the robot with a Ideal-Vr safety configuration. However, all of them felt safe working with Real-Vr, Real-Vo, Ideal-Vo, and LiDAR-SM safety configurations. The perception of Ideal-Vr and Real-Vr configurations for the TriSSM-Vr by the human subject are contradictory. One of the reasons for this is that the *Real* sensors are more responsive and sensitive to minimum distance change resulting in better anticipation and predictability.

According to the response of *Q3*, the highest average rating in-terms of predictability, anticipation or intuitiveness of the robot behavior is for Real-Vr. On average, all the SSM safety configurations were predictable for the human subjects.

On average, 66% of the human subjects found the visual feedback from the LED indicators helpful during all SSM safety configurations, as inferred from the response of *Q4*. The sound of the robot motors was mentioned by some of the human subjects as better feedback to determine robot behavior. Another observation noted by the human subjects was that the LED indicators were not always visible, and better placement that would allow them to be present in the periphery of the human vision while performing the tasks was suggested. Another suggestion was to make the LED indicators have varying flickers and intensity changes to make them visibly distinct.

From the response to *Q5*, 23% of the human subjects for the Real-SM SSM configuration, did not feel comfortable after familiarizing themselves with the robot motion. The response to the rating of comfort level from the human subjects in *Q6* suggested an overall high score. The least comfortable for the human subjects was Ideal-Vr, and Real-Vo being the most comfortable.

At the end of the experiment, the human subject was asked if they would work with the same robot behavior. As per the responses for *Q7*, the human-subjects agreed completely to work with Real-Vo, Real-Vr, and Ideal-Vo SSM configurations. All LiDAR-based configurations resulted in over 14% of human subjects being doubtful working with it. For Real-SM, over 30% of the human subjects were either unsure or did not prefer working with the Real-SM configuration.

A one-tailed (or one-sided) pair-wise t-test [126] for the two-sample populations assuming unequal variances was performed to measure the statistical significance of the differences in the human-subject responses to the questionnaire for the TriSSM-Vo and TriSSM-Vr SSM safety configurations. The statistics and the resulting p-value from the t-test are shown in Figure 5.13. It was observed that there was some evidence ($p \approx 0.05$) that the human subjects felt safer working with TriSSM-Vo than TriSSM-Vr based safety configurations.

To summarize, Real-Vo, Real-Vr, Ideal-Vo, and Sim-Vo were the most favorable of SSM safety configurations in terms of human emotional state, safety, comfort, and preference to work with. LED indicators were considered helpful as visual feedback by over third of all the human subjects. It could be more helpful with better placement of the LED indicators in the workspace.

In addition to the questionnaires, the human subjects also rated their experience during the experiments. The results are presented in the following section.

5.2.3.2 Overall Ratings for the Dynamic SSM setups

The human subject rated the experiments performed during a session in terms of best preference to work with, overall perceived speed of the robot, their level of safety/comfort, and predictable or anticipatory nature of the robot. A Likert scale (a scale of *1 to 5*, with *5* being the highest) was used to record the ratings. The questionnaires recorded responses

Safety Config	Statistics	Q1 No Anxiety, Calmness & No Surprise	Q2 Feel Safe? (Y/N)	Q3 How intiutive/ anticipatory?	Q5 Feel comfortable? (Y/N)	Q6 How comfortable?	Q7 Feel safe working again? (Y/N)
TriSSM-Vo*	AVG	**3.437**	**1.000**	3.738	**0.976**	4.429	**1.000**
	STD-S	0.777	0.000	0.828	0.154	0.590	0.000
	STD-ERROR	0.120	0.000	0.128	0.024	0.091	0.000
	MEDIAN	3.500	1.000	4.000	1.000	4.000	1.000
TriSSM-Vr*	AVG	3.250	0.893	**3.821**	0.929	**4.464**	0.929
	STD-S	0.976	0.315	0.819	0.262	0.576	0.262
	STD-ERROR	0.184	0.060	0.155	0.050	0.109	0.050
	MEDIAN	3.667	1.000	4.000	1.000	4.500	1.000

TriSSM-Vo* (vs) Vr*		Q1	Q2	Q3	Q5	Q6	Q7
	p-value	0.200288	**0.041522**	0.339753	0.195852	0.401122	0.080520
	Significance (alpha)	0.05	0.05	0.05	0.05	0.05	0.05
	Mean-Differnce (p-value < alpha)	NOT SIGNIFICANT	**SIGNIFICANT**	NOT SIGNIFICANT	NOT SIGNIFICANT	NOT SIGNIFICANT	NOT SIGNIFICANT
	Observation	Vo > Vr	**Vo > Vr**	Vr > Vo	Vo > Vr	Vr > Vo	Vo > Vr

* For Real, Ideal, and Sim Sensor Modalities

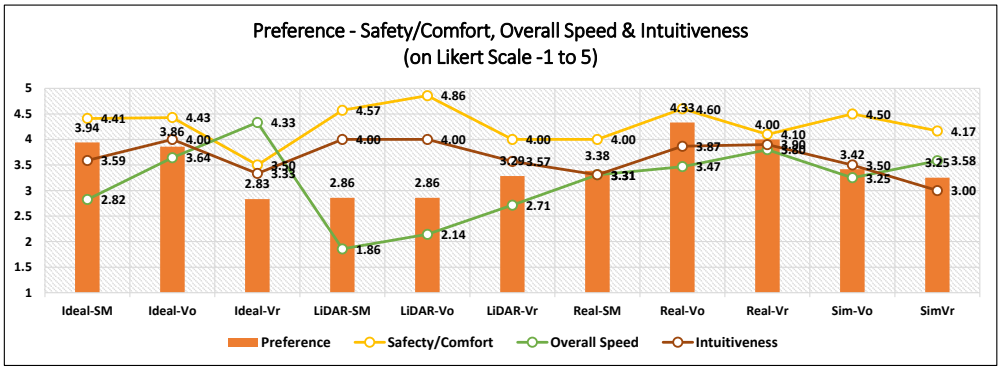
Figure 5.11: The statistics and the resulting p-values from a t-test to measure the statistical significance of the differences in the human-subject responses to the questionnaire for TriSSM-Vo and TriSSM-Vr safety configurations.

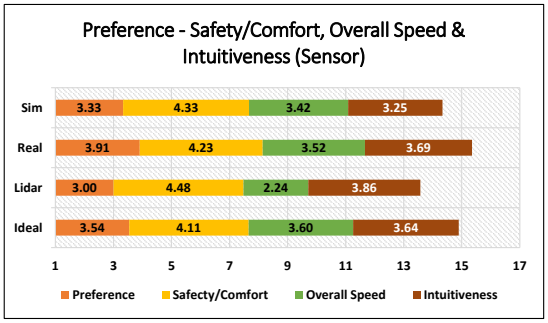
from the human subject following each experiment, while the ratings provided a comparison of robot behavior for all the SSM safety configurations during a session by the human subject. During the session, the baseline experiment with the robot was also performed to provide the human subject with a reference robot behavior to compare with.

The summarized results of the ratings of all the dynamic SSM safety configurations as rated by the 19 human subjects are shown in Figure 5.12. The ratings are averaged for comparison of the safety algorithm and the sensor modalities. It is observed that on average, Real-Vo is the most preferred SSM configuration, and LiDAR SM, LiDAR-Vo, and Ideal-Vr are the least preferred SSM configurations. In terms of safety and comfort, all the SSM configurations are rated high on average; LiDAR-Vo and Real-Vo are rated the highest. The highest overall perceived speed is for Ideal-Vr, and the lowest is for LiDAR-SM. It is observed that for all sensor modalities the overall perceived speed can be ordered as TriSSM-Vr > TriSSM-Vo > Tri-SM. In terms of intuitiveness Ideal-Vo, LiDAR-SM, and LiDAR-Vo are rated higher than the rest. In terms of SSM algorithm for all sensor modalities, the ratings are ordered as TriSSM-Vo > Tri-SM > TriSSM-Vr.

To summarize, Real-Vo and Ideal-Vo are the highest rated SSM safety configurations.

Sensor-Algo	Preference			Safety/Comfort			Overall Speed			Intuitiveness		
	AVG	STD-ERROR	STD-S	AVG	STD-ERROR	STD-S	AVG	STD-ERROR	STD-S	AVG	STD-ERROR	STD-S
Ideal-SM	3.941	0.201	0.827	4.412	0.173	0.712	2.824	0.231	0.951	3.588	0.228	0.939
Ideal-Vo	3.857	0.275	1.027	4.429	0.228	0.852	3.643	0.289	1.082	***4.000***	0.210	0.784
Ideal-Vr	2.833	0.477	1.169	3.500	0.428	1.049	***4.333***	0.333	0.816	3.333	0.422	1.033
LiDAR-SM	2.857	0.340	0.900	4.571	0.297	0.787	1.857	0.340	0.900	***4.000***	0.436	1.155
LiDAR-Vo	2.857	0.261	0.690	***4.857***	0.143	0.378	2.143	0.340	0.900	***4.000***	0.309	0.816
LiDAR-Vr	3.286	0.565	1.496	4.000	0.218	0.577	2.714	0.474	1.254	3.571	0.528	1.397
Real-SM	3.385	0.368	1.325	4.000	0.300	1.080	3.308	0.365	1.316	3.308	0.398	1.437
Real-Vo	***4.333***	0.211	0.816	4.600	0.131	0.507	3.467	0.350	1.356	3.867	0.236	0.915
Real-Vr	***4.000***	0.394	1.247	4.100	0.233	0.738	3.800	0.327	1.033	3.900	0.233	0.738
Sim-Vo	3.417	0.313	1.084	4.500	0.195	0.674	3.250	0.329	1.138	3.500	0.230	0.798
SimVr	3.250	0.429	1.485	4.167	0.322	1.115	3.583	0.336	1.165	3.000	0.213	0.739





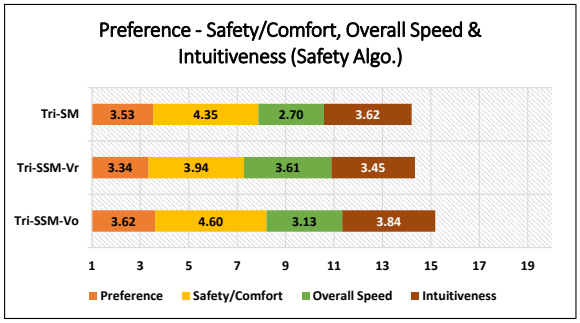


Figure 5.12: (Top) The tabulated average rating metrics of experiments as perceived by the human subjects. (Center) The graph shows the mean and std. preference, safety/comfort, overall speed, and intuitiveness as perceived by the human subjects to varying safety setups of sensors and SSM safety algorithms. (Bottom) The graphs consolidate the comparison between sensors and safety algorithms.

The overall robot perceived speed is slower for LiDAR based SSM configuration in comparison to other sensor modalities. The lowest rated safety configurations are Ideal-Vr and LiDAR-SM due to its lower preference to work with and slow overall speed. These observations align with the objective evaluation in Section 5.2.2.

A. Comparing TriSSM-Vo and TriSSM-Vr safety approaches

In this section, the claim 'TriSSM-Vo based safety approach is safer, more anticipatory (predictable) and comfortable for the human agent to work with, than TriSSM-Vr' is corroborated. For corroboration, the safety/comfort and anticipatory/predictable ratings are analyzed. The comparison is done for the observations of *Real*, *Ideal*, and *Sim* based TriSSM-Vo and TriSSM-Vr safety configurations. Let the population means of the safety/comfort rating for TriSSM-Vo and TriSSM-Vr be μ^{Vo}_{safe} and μ^{Vr}_{safe}, respectively. The null and the alternative hypotheses representing the claim comparing the safety of the SSM safety configuration can be written as

$\mathbf{H^s_0}$: The mean safety/comfort rating for TriSSM-Vo (μ^{Vo}_{safe}) is equal to the mean safety rating of TriSSM-Vr (μ^{Vr}_{safe}) for the *Real*, *Ideal*, and *Sim* based sensor modalities.

$\mathbf{H^s_a}$: The mean safety/comfort rating for TriSSM-Vo (μ^{Vo}_{safe}) is greater than the mean safety rating of TriSSM-Vr (μ^{Vr}_{safe}) for the *Real*, *Ideal*, and *Sim* based sensor modalities.

$$\begin{aligned} H^s_0 &: \mu^{Vo}_{safe} = \mu^{Vr}_{safe}, \\ H^s_a &: \mu^{Vo}_{safe} > \mu^{Vr}_{safe} \end{aligned} \tag{5.1}$$

Similarly, for the population means of the anticipatory/predictable ratings for TriSSM-Vo and TriSSM-Vr, say μ^{Vo}_{pred} and μ^{Vr}_{pred}, respectively; the null and the alternative hypotheses representing the claim comparing the anticipatory nature or predictability of the SSM configuration can be written as

$\mathbf{H^p_0}$: The mean anticipatory/predictable rating for TriSSM-Vo (μ^{Vo}_{pred}) is equal to the mean anticipatory/predictable rating of TriSSM-Vr (μ^{Vr}_{pred}) for the *Real*, *Ideal*, and *Sim* based sensor modalities.

$\mathbf{H^p_a}$: The mean anticipatory/predictable rating for TriSSM-Vo (μ^{Vo}_{pred}) is greater than the mean anticipatory/predictable rating of TriSSM-Vr (μ^{Vr}_{pred}) for the *Real*, *Ideal*, and *Sim* based sensor modalities.

Safety Config	Rating	AVG	STD-S	STD-ERROR	MEDIAN
TriSSM-Vo*	Preference	3.902	1.020	0.159	4.000
	Safety/Comfort	**4.512**	0.675	0.105	5.000
	Percieved Speed	3.463	1.185	0.185	3.000
	Anticipatory/Predictable	**3.805**	0.843	0.132	4.000
	Overall	**3.137**	0.450	0.070	3.200
TriSSM-Vr*	Preference	3.429	1.372	0.259	4.000
	Safety/Comfort	4.000	0.981	0.185	4.000
	Percieved Speed	3.821	1.056	0.200	4.000
	Anticipatory/Predictable	3.393	0.875	0.165	3.000
	Overall	2.929	0.537	0.101	2.800

Comparison	Rating	p-value	Significance (alpha)	Mean Difference (p-value<alpha)	Observation
TriSSM-Vo* (vs) TriSSM-Vr*	Preference	0.063126	0.05	NOT SIGNIFICANT	TriSSM-Vo > TriSSM-Vr
	Safety/Comfort	0.010318	0.05	SIGNIFICANT	**TriSSM-Vo > TriSSM-Vr**
	Percieved Speed	0.096616	0.05	NOT SIGNIFICANT	TriSSM-Vr > TriSSM-Vo
	Anticipatory/Predictable	0.028128	0.05	SIGNIFICANT	**TriSSM-Vo > TriSSM-Vr**
	Overall	0.049012	0.05	SIGNIFICANT	**TriSSM-Vo > TriSSM-Vr**

* For ratings of Real, Ideal, and Sim Sensor Modalities

Figure 5.13: The statistics and the resulting p-values from a t-test to measure the statistical significance of the differences in means of the ratings for TriSSM-Vo and TriSSM-Vr safety configurations.

$$H_0^p : \mu_{pred}^{Vo} = \mu_{pred}^{Vr}, \\ H_a^p : \mu_{pred}^{Vo} > \mu_{pred}^{Vr} \tag{5.2}$$

In order to measure the statistical significance of the differences of the safety/comfort and anticipatory/predictable mean for the TriSSM-Vo and TriSSM-Vr SSM safety configurations, a one-tailed (or one-sided) pair-wise t-test [126] for the two-sample populations assuming unequal variances was performed. The statistics and the resulting p-value from the t-test are shown in Figure 5.13.

It can be observed that there is a statistically significant ($p < 0.05$) evidence that the human subject felt safer and more comfortable during the TriSSM-Vo safety configuration than TriSSM-Vr for the *Real*, *Ideal* and *Sim* based sensor modalities. It is also observed that there is statistically significant ($p < 0.05$) evidence that the robot behavior is more anticipatory/predictable during TriSSM-Vo than TriSSM-Vr safety configuration.

Safety Config	Rating	AVG	STD-S	STD-ERROR	MEDIAN
TriSSM-Vo*	Preference	**4.103**	0.939	0.174	4.000
	Safety/Comfort	**4.517**	0.688	0.128	5.000
	Percieved Speed	3.552	1.213	0.225	3.000
	Anticipatory/Predictable	**3.931**	0.842	0.156	4.000
	Overall	**3.221**	0.445	0.083	3.200
TriSSM-Vr*	Preference	3.563	1.315	0.329	4.000
	Safety/Comfort	3.875	0.885	0.221	4.000
	Percieved Speed	**4.000**	0.966	0.242	4.000
	Anticipatory/Predictable	3.688	0.873	0.218	4.000
	Overall	3.025	0.536	0.134	2.900
Tri-SM*	Preference	3.700	1.088	0.199	4.000
	Safety/Comfort	4.233	0.898	0.164	4.500
	Percieved Speed	3.033	1.129	0.206	3.000
	Anticipatory/Predictable	3.467	1.167	0.213	4.000
	Overall	2.887	0.567	0.104	3.000

Comparison	Rating	p-value	Significance (alpha)	Mean Difference (p-value<alpha)	Observation
TriSSM-Vo* (vs) Tri-SM*	Preference	0.066213	0.05	NOT SIGNIFICANT	TriSSM-Vo > Tri-SM
	Safety/Comfort	0.088712	0.05	NOT SIGNIFICANT	TriSSM-Vo > Tri-SM
	Percieved Speed	0.047503	0.05	**SIGNIFICANT**	**TriSSM-Vo > Tri-SM**
	Anticipatory/Predictable	0.042327	0.05	**SIGNIFICANT**	**TriSSM-Vo > Tri-SM**
	Overall	0.007332	0.05	**SIGNIFICANT**	**TriSSM-Vo > Tri-SM**
TriSSM-Vr* (vs) Tri-SM*	Preference	0.361601	0.05	NOT SIGNIFICANT	Tri-SM > TriSSM-Vr
	Safety/Comfort	0.101339	0.05	NOT SIGNIFICANT	Tri-SM > TriSSM-Vr
	Percieved Speed	0.002198	0.05	**SIGNIFICANT**	**TriSSM-Vr > Tri-SM**
	Anticipatory/Predictable	0.236679	0.05	NOT SIGNIFICANT	TriSSM-Vr > Tri-SM
	Overall	0.210032	0.05	NOT SIGNIFICANT	TriSSM-Vr > Tri-SM

* For ratings of Real and Ideal Sensor Modalities

Figure 5.14: The statistics and the resulting p-values from a t-test to measure the statistical significance of the differences in means of the ratings for TriSSM-Vo and Tri-SM, and TriSSM-Vr and Tri-SM safety configurations.

In order to compare the ratings of TriSSM-Vo and TriSSM-Vr with Tri-SM safety configurations, a similar one-tailed pair-wise t-test was done. The statistics and the resulting p-value from the t-test are shown in Figure 5.14. The safety configurations were limited to the observations of *Real* and *Ideal* sensor modalities, as the number of observations for *Sim* based Tri-SM was very small.

It can be observed that there is a statistically significant ($p < 0.05$) evidence that the human subject felt that the robot motion during TriSSM-Vr was faster than Tri-SM safety configuration. It is also observed that there is some ($p \approx 0.05$) evidence that the robot

motion is faster, and the robot behavior is more anticipatory/predictable during TriSSM-Vo than Tri-SM safety configuration. However, overall there is significant evidence that the TriSSM-Vo is rated higher than Tri-SM.

Next, we corroborate another claim comparing the dynamic Tri-SSM safety configurations with *Real* and *Ideal* sensor modalities.

B. Comparing Real and Ideal based dynamic Tri-SSM safety configurations

In this section, the claim 'the minimum separation distance error in the 8-node ToF sensor array (*Real*) compared to the motion capture (*Ideal*) based Tri-SSM safety configurations, affects how safe or comfortable the human subject feels' during an HRC task, is corroborated. For corroboration, the safety/comfort ratings are analyzed.

Let the population means of the safety/comfort rating for *Real* and *Ideal*-based safety configurations be μ_{safe}^{Real} and μ_{safe}^{Ideal}, respectively. The null and the alternative hypotheses representing the claim comparing the safety of the SSM safety configuration can be written as

$\mathbf{H_0^s}$: The mean safety/comfort rating for *Real* (μ_{safe}^{Real}) is equal to the mean safety rating of *Ideal* (μ_{safe}^{Ideal}) based dynamic Tri-SSM configurations.

$\mathbf{H_a^s}$: The mean safety/comfort rating for *Real* (μ_{safe}^{Real}) is not equal to the mean safety rating of *Ideal* (μ_{safe}^{Ideal}) based dynamic Tri-SSM configurations.

$$\begin{aligned} H_0^s &: \mu_{safe}^{Real} = \mu_{safe}^{Ideal}, \\ H_a^s &: \mu_{safe}^{Real} \neq \mu_{safe}^{Ideal} \end{aligned} \tag{5.3}$$

A two-tailed (or two-sided) pair-wise *t*-test [126] for the two-sample populations assuming unequal variances was performed. This test measured if there is statistically significant evidence that the mean of the safety/comfort rating for *LiDAR*-based Tri-SSM configurations is not equal to *Ideal*-based configurations. In addition, a similar two-tailed (or two-sided) pair-wise *t*-test was performed for all the other ratings as well. The statistics and the resulting *p*-value from the *t*-test are shown in Figure 5.15.

It is observed that there is no statistically significant evidence ($p > 0.05$) that the human-subject feels less safe/comfortable during the HRC task with *Real* sensor modality based Tri-SSM than *Ideal*-based Tri-SSM configurations. Moreover, it is observed that

all the ratings are approximately equal for *Real* and *Ideal*-based Tri-SSM safety configurations.

A one-tailed (or one-sided) pair-wise *t*-test [126] was done to compare the ratings of *Real* and *Ideal* with 2D *LiDAR*-based Tri-SSM configurations. The results are shown in Figure 5.15. It is observed that there is statistically significant evidence that the human subjects felt the robot motion was faster and the robot behavior was preferable for *Real* and *Ideal*, than 2D *LiDAR*-based Tri-SSM safety configurations.

Next, we present the observations of the assembly task performed by the human-operator with different SSM safety configurations.

5.2.3.3 Observations of the Human Assembly Task

In this experiment, the robot aids the human by supplying the part (*Part-2*) for assembly. The human has four parts placed in a bin as a head start. The desired scenario during this task is that at the end of the experiment, the human should have assembled ten parts or less. In other words, the robot is able to provide a *Part-2* at the same or faster rate as the human picks *Part-2* for assembly. This can be quantified as the ratio of the maximum robot supplied parts during the experiment (10 parts) to the sum of assembled parts by the human ($n_{assembly}$) and the parts yet to be supplied (the robot lag n_{lag}) when the robot is unable to keep up with the human assembly rate (there are no *Part-2* available for assembly to the human). This quantity is termed as the *human and robot matching efficiency* (HRE). Let HRE be $\mathcal{E}$, and can be defined as

$$\begin{aligned} \textit{Human-Robot Matching Efficiency, } HRE &= \frac{\textit{Robot Supplied Parts}}{\textit{Assembled Parts} + \textit{RobotLag}} \\ \text{or} \\ \mathcal{E} &= \frac{10}{n_{assembly} + n_{lag}} \end{aligned} \tag{5.4}$$

The summarized results of these observations are shown in Figure 5.16. It can be observed that Real-Vo and Ideal-Vr are the only SSM configurations where the robot did not lag behind. Whereas, in the LiDAR-based SSM configurations, the robot consistently lagged. On average, when the robot had four parts left on the pallet for pick up, the human had completed ten assemblies, and has to wait for the remaining parts to be supplied.

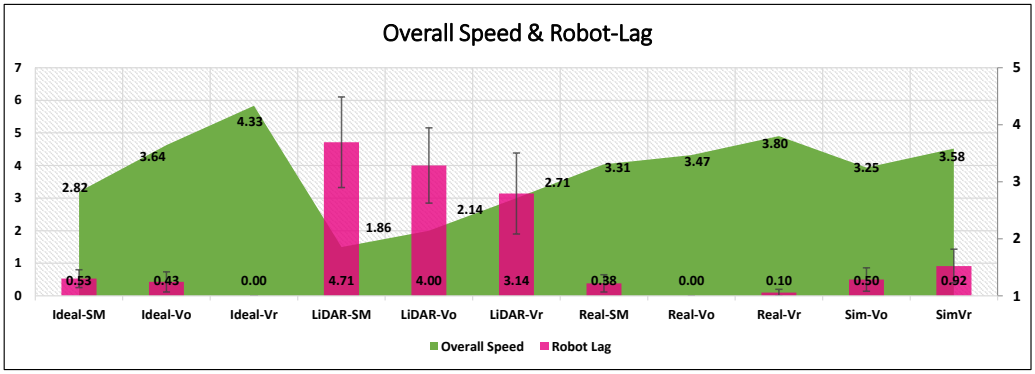
Safety Config	Rating	AVG	STD-S	STD-ERROR	MEDIAN
Real-*	Preference	3.921	1.156	0.187	4.000
	Safety/Comfort	4.263	0.817	0.133	4.000
	Percieved Speed	3.500	1.230	0.200	3.500
	Anticipatory/Predictable	3.684	1.079	0.175	4.000
	Overall	3.842	0.757	0.123	4.000
Ideal-*	Preference	3.730	1.004	0.165	4.000
	Safety/Comfort	4.270	0.859	0.141	5.000
	Percieved Speed	3.378	1.099	0.181	3.000
	Anticipatory/Predictable	3.703	0.896	0.147	4.000
	Overall	3.770	0.537	0.088	3.750
LiDAR-*	Preference	3.000	1.049	0.229	3.000
	Safety/Comfort	4.476	0.680	0.148	5.000
	Percieved Speed	2.238	1.044	0.228	2.000
	Anticipatory/Predictable	3.857	1.108	0.242	4.000
	Overall	3.393	0.683	0.149	3.500

Comparison	Rating	p-value	Significance (alpha)	Mean Difference (p-value<alpha)	Observation
Real-* (vs) Ideal-*	Preference	0.452323	0.05	NOT SIGNIFICANT	Real* ≈Ideal*
	Safety/Comfort	0.971198	0.05	NOT SIGNIFICANT	Real* ≈Ideal*
	Percieved Speed	0.657131	0.05	NOT SIGNIFICANT	Real* ≈Ideal*
	Anticipatory/Predictable	0.936658	0.05	NOT SIGNIFICANT	Real* ≈Ideal*
	Overall	0.641017	0.05	NOT SIGNIFICANT	Real* ≈Ideal*
Real-* (vs) LiDAR-*	Preference	0.003353	0.05	SIGNIFICANT	Real* > LIDAR*
	Safety/Comfort	0.292295	0.05	NOT SIGNIFICANT	LIDAR* > Real*
	Percieved Speed	0.000139	0.05	SIGNIFICANT	Real* > LIDAR*
	Anticipatory/Predictable	0.567364	0.05	NOT SIGNIFICANT	LIDAR* > Real*
	Overall	0.025238	0.05	SIGNIFICANT	Real* > LIDAR*
Ideal-* (vs) LiDAR-*	Preference	0.013804	0.05	SIGNIFICANT	Ideal* > LIDAR*
	Safety/Comfort	0.322577	0.05	NOT SIGNIFICANT	LIDAR* > Ideal*
	Percieved Speed	0.000325	0.05	SIGNIFICANT	Ideal* > LIDAR*
	Anticipatory/Predictable	0.590376	0.05	NOT SIGNIFICANT	LIDAR* > Ideal*
	Overall	0.036846	0.05	SIGNIFICANT	Ideal* > LIDAR*

* For all TriSSM-Vo, TriSSM-Vr and Tri-SM saftey configurations.

Figure 5.15: The statistics and the resulting p-values from a t-test to measure the statistical significance of the differences in means of the ratings for *Real*, *Ideal* and 2D *LiDAR*-based Tri-SSM safety configurations.

Sensor-Algo	Assembled Parts (Max 14)			Robot Lag						Human Robot-Matching Efficiency(%)		
	AVG	STD-ERROR	STD-S	AVG	STD-ERROR	STD-S	Count-Lag	ExpCount	CountLag-Ratio	AVG	STD-ERROR	STD-S
Ideal-SM	12.00	0.364	1.500	0.529	0.273	1.125	4	17	0.235	0.824	0.035	0.144
Ideal-Vo	11.00	0.565	2.112	0.429	0.309	1.158	2	14	0.143	0.923	0.056	0.210
Ideal-Vr	11.67	0.760	1.862	0.000	0.000	0.000	0	6	0.000	0.875	0.056	0.138
LiDAR-SM	***13.29***	0.474	1.254	***4.714***	1.392	3.684	5	7	0.714	0.599	0.073	0.193
LiDAR-Vo	***13.00***	0.655	1.732	***4.000***	1.155	3.055	5	7	0.714	0.639	0.083	0.220
LiDAR-Vr	12.71	0.747	1.976	3.143	1.243	3.288	4	7	0.571	0.694	0.093	0.245
Real-SM	11.38	0.460	1.660	0.385	0.266	0.961	2	13	0.154	0.879	0.044	0.160
Real-Vo	10.87	0.456	1.767	0.000	0.000	0.000	0	15	0.000	0.944	0.040	0.156
Real-Vr	11.00	0.516	1.633	0.100	0.100	0.316	1	10	0.100	0.921	0.044	0.138
Sim-Vo	11.92	0.514	1.782	0.500	0.359	1.243	2	12	0.167	0.836	0.046	0.158
SimVr	11.42	0.529	1.832	0.917	0.514	1.782	3	12	0.250	0.860	0.057	0.196



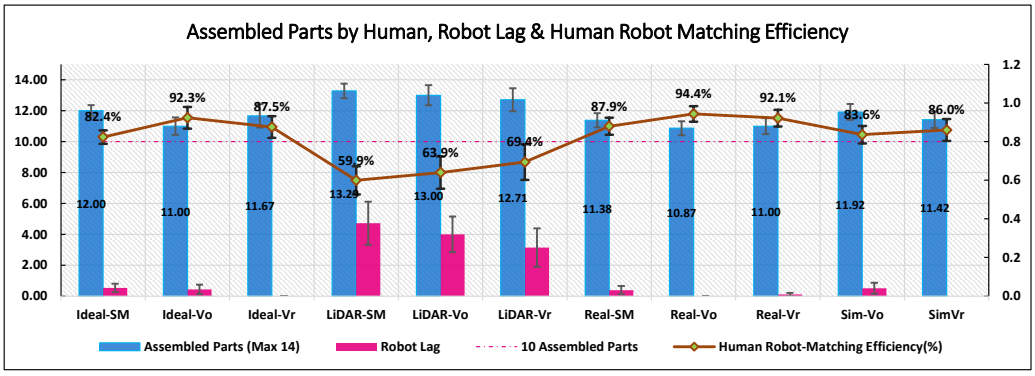


Figure 5.16: (Top) The tabulated average assembled parts and the times robot lagged behind to the human to assist in the completion of the task. (Center) The mean robot lag for all the SSM safety configurations, in comparison to the perceived overall speed. (Bottom) The average number of parts assembled by the human subject, the average robot lag and the average human-robot matching efficiency for all the SSM Safety Configurations.

Next, the claim that 'the least productive Tri-SM based safety configuration using the *Real* (ToF) and *Ideal* (Mocap) based sensor modalities is more productive than all the SSM safety configurations using 2D *LiDAR*,' is corroborated. For corroboration, the *productivity* or *efficiency* of the HRC task is analyzed. The comparison of the productivity or efficiency of the HRC task for a dynamic SSM safety configuration is done based on the significance of the differences in the means ($\mu_{\mathcal{E}}, \mu_{lag}$) of the human-robot matching efficiency ($\mathcal{E}$) and the number of parts the robot lagged behind (n_{lag}), respectively.

The null and the alternative hypotheses representing the claim can be written as

$\mathbf{H_0^{1,2}}$: The mean human-robot matching efficiency (HRE, $\mathcal{E}$) of the robot *Real* ($\mu_{\mathcal{E}}^{Real}$) and *Ideal* ($\mu_{\mathcal{E}}^{Ideal}$) based Tri-SM configurations is equal to the mean HRE of the robot with all the 2D *LiDAR* ($\mu_{\mathcal{E}}^{Lidar}$) based TriSSM (Vo,Vr and SM) configurations.

$\mathbf{H_a^{1,2}}$: The mean human-robot matching efficiency (HRE, $\mathcal{E}$) of the robot with *Real* ($\mu_{\mathcal{E}}^{Real}$) and *Ideal* ($\mu_{\mathcal{E}}^{Ideal}$) based Tri-SM configurations is greater than the mean HRE of the robot with all the 2D *LiDAR* ($\mu_{\mathcal{E}}^{Lidar}$) based TriSSM (Vo,Vr and SM) configurations.

$$\begin{aligned} H_0^1 :\ \mu_{\mathcal{E}}^{Lidar} = \mu_{\mathcal{E}}^{Real}, H_a^1 :\ \mu_{\mathcal{E}}^{Lidar} < \mu_{\mathcal{E}}^{Real} \\ H_0^2 :\ \mu_{\mathcal{E}}^{Lidar} = \mu_{\mathcal{E}}^{Ideal}, H_a^2 :\ \mu_{\mathcal{E}}^{Lidar} < \mu_{\mathcal{E}}^{Ideal} \end{aligned} \tag{5.5}$$

and

$\mathbf{H_0^{3,4}}$: The mean number of parts the robot lagged behind (n_{lag}) with *Real* (μ_{lag}^{Real}) and *Ideal* (μ_{lag}^{Ideal}) based Tri-SM configurations is equal to the mean n_{lag} with all the 2D *LiDAR* (μ_{lag}^{Lidar}) based TriSSM (Vo, Vr, and SM) configurations.

$\mathbf{H_a^{3,4}}$: The mean number of parts the robot lagged behind (n_{lag}) with *Real* (μ_{lag}^{Real}) and *Ideal* (μ_{lag}^{Ideal}) based Tri-SM configurations is less than the mean n_{lag} with all the 2D *LiDAR* (μ_{lag}^{Lidar}) based TriSSM (Vo, Vr, and SM) configurations.

$$\begin{aligned} H_0^3 :\ \mu_{lag}^{Lidar} = \mu_{lag}^{Real}, H_a^3 :\ \mu_{lag}^{Lidar} > \mu_{lag}^{Real} \\ H_0^4 :\ \mu_{lag}^{Lidar} = \mu_{lag}^{Ideal}, H_a^4 :\ \mu_{lag}^{Lidar} > \mu_{lag}^{Ideal} \end{aligned} \tag{5.6}$$

A one-tailed pair-wise *t*-test for the two-sample populations assuming unequal variances was performed. This test measured if there was any statistically significant evidence in the differences of the means of HRE and the robot lag, for all the SSM configurations using a 2D *LiDAR*, and the *Real* and *Ideal* sensor modalities. The statistics and the resulting

	Human-Robot Efficiency (Productivity Criteria)			
Safety Config.	**AVG**	**STD-S**	**STD-ERROR**	**MEDIAN**
LiDAR-(Vo,Vr,SM)	0.644	0.213	0.046	0.526
Real-SM	0.879	0.160	0.044	0.909
Ideal-SM	0.824	0.144	0.035	0.833

Comparison	p-value	Significance(alpha)	Mean-Difference (p-value<alpha)	Observation
RealSM vs LiDAR*	0.000470	0.05	**SIGNIFICANT**	**RealSM> LiDAR***
IdealSM vs LiDAR*	0.001977	0.05	**SIGNIFICANT**	**IdealSM> LiDAR***

* For LiDAR based TriSSM-Vo, TriSSM-Vr and Tri-SM safety configurations

(a)

	Robot Lagging Behind			
Safety Config.	**AVG**	**STD-S**	**STD-ERROR**	**MEDIAN**
LiDAR-(Vo,Vr,SM)	3.952	3.248	0.709	5.000
Real-SM	0.385	0.961	0.266	0.000
Ideal-SM	0.529	1.125	0.273	0.000

Comparison	p-value	Significance(alpha)	Mean-Difference (p-value<alpha)	Observation
RealSM vs Lidar*	0.000039	0.05	**SIGNIFICANT**	**LiDAR*>RealSM**
IdealSM vs Lidar*	0.000063	0.05	**SIGNIFICANT**	**LiDAR*> IdealSM**

* For LiDAR based TriSSM-Vo, TriSSM-Vr and Tri-SM safety configurations

(b)

Figure 5.17: The statistics and the resulting p-value from a t-test to measure the statistical significance of the differences in the HRE (a) and the number of parts robot lagged behind (b), for all the SSM configurations using a 2D *LiDAR* and the Tri-SM safety configurations using *Real* and *Ideal* sensor modalities.

p-value from the t-test are shown in Figure 5.17(a) and Figure 5.17(b).

It can be observed that there is a statistically significant ($p < 0.05$) evidence that the HRE of the *Real* and *Ideal*-based Tri-SM safety configuration is higher than all the *LiDAR*-based SSM configurations. It was also observed that there is statistically highly significant ($p < 0.001$) evidence that the number of parts the robot lags behind during *LiDAR*-based SSM safety configurations is greater than *Real* and *Ideal*-based Tri-SM safety configurations.

Lastly, in the next section, we discuss the results of human-subject gaze tracking during the experiments.

	LED Indicator	Robot Shoulder	Robot Elbow	Robot Tool	Table-0	Table-1	Table-2	Table-3	Table-4
Ideal-SM	6.285	14.505	20.470	17.049	2.962	17.287	14.432	30.659	5.493
Ideal-Vo	5.849	12.940	15.276	14.644	4.154	18.992	12.914	29.429	6.991
Ideal-Vr	4.738	14.362	17.420	15.536	1.102	14.836	15.056	33.882	3.448
LiDAR-SM	9.584	27.246	35.824	30.536	7.182	35.492	27.160	44.740	10.790
LiDAR-Vo	7.423	19.568	27.804	25.938	5.229	22.644	20.343	32.511	5.403
LiDAR-Vr	5.894	15.433	19.719	16.790	4.067	21.667	18.417	36.989	5.661
Real-SM	6.029	15.133	20.461	21.726	4.549	19.104	17.444	30.236	5.730
Real-Vo	4.748	11.146	16.443	16.082	3.505	15.188	11.835	28.359	4.794
Real-Vr	4.619	12.039	14.418	11.751	1.781	12.034	11.434	23.570	3.699
Sim-Vo	3.413	13.040	16.923	14.709	4.263	12.608	12.305	28.376	3.425
Sim-Vr	5.371	11.674	14.340	13.431	2.094	15.100	11.395	28.026	5.666
	LED Indicator	Robot Shoulder	Robot Elbow	Robot Tool	Table-0	Table-1	Table-2	Table-3	Table-4
Average Gaze Ti	5.81	15.19	19.92	18.02	3.72	18.63	15.70	31.53	5.55
Std-Error	0.47	1.33	1.88	1.65	0.49	1.89	1.40	1.61	0.59

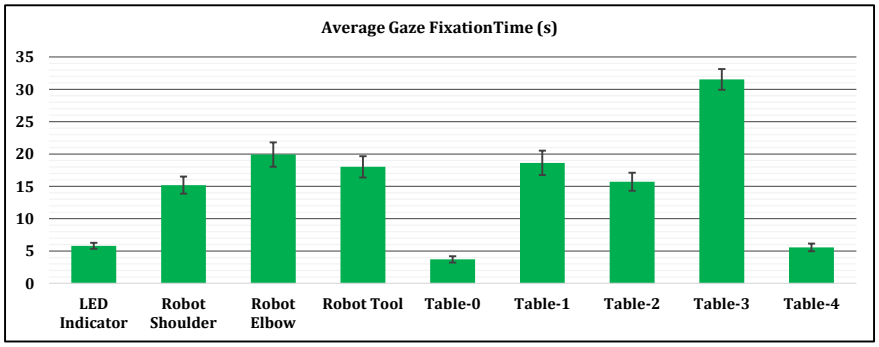


Figure 5.18: (Top) The tabulated average gaze fixation times calculated using Pupil Eye glasses and V-REP.(Bottom) The graph shows the mean and std. error showing the objects in the shared workspace that the human gaze is fixated while performing the task.

5.2.3.4 Gaze Tracking during HRC Experiment

In order to quantify the attention of the human subject on the objects of interest(*Tables*, robot links, and LED indicator) in the workspace during the experiment, the gaze tracking and the duration of gaze-fixation information is used [67]. The results of the average gaze fixation times on objects of interest are calculated using the Pupil-Labs Eyeglasses, and a digital-twin implementation in V-REP are shown in Figure 5.18. It is observed that the maximum fixation is for *Table-3* where the human-subject performs the *screw-in* task step during the assembly. This indicates that this step requires more time and hand-eye coordination. Similarly, *Table-1* and *Table-2*, for the picking and *drop-in* steps require the human attention in order to complete the assembly.

The robot operating workspace overlaps with the human motion during the task, resulting in the human being more aware of the robot. Hence, the human-subject also looks at the robot links during the task, especially the elbow and the tool links. The LED indicators are focused less in comparison to the robot links by the human-subject. This could be either

because the LED indicators were not placed in the periphery of the human vision or the human-subject preferred the robot links and sound of motors to predict and be aware of the robot actions/operation mode.

Next, we shed light on the limitations of this HRC experiment setup in the following section.

5.3 Limitations

The number of human-subject experiments is limited to 19 subjects, and more experiments are needed. These results are specific to a standard robot pick and place and a human assembly task, which presents a human-robot coexistence scenario of human-robot collaboration (refer Section 2.2). The results for this task conclusively indicate that a dynamic SSM setup using on-robot ToF sensor arrays and motion-capture is more productive and safe compared to a 2D LiDAR-based setup, conventionally used in the industry. However, these results need to be validated and investigated in varying HRC scenarios such as hand-over tasks, hand guiding, and shared/cooperation tasks (refer Section 2.2) [15, 17].

The objective of the human-subject experiments was not to create a mental model for understanding and predicting the human or robot behavior during an HRC task, but to evaluate and validate the performance of an on-robot ToF sensor technology developed to enable safe human-robot interaction. The experiments did not account for the human factors or the human experiences that may change by changing parameters such as the distance thresholds and robot speeds of an SSM configuration. In this experiment, the human-subject study was similar to a 'report card' or review of SSM safety configurations defining the rule-based robot behavior, and not on how the various features of each safety configuration can affect the human behavior during HRC.

Moreover, these experiments were performed in a controlled laboratory research setting and not in the field or an industrial environment with experienced human-operators that work with industrial robots or machinery. The duration of each experiment was shorter (3-4 minutes) compared to that in industry (continuous and can be for hours). In an industrial setting, human factors such as physical and mental fatigue, or environmental factors such as pollution or machinery noise, could also affect the human subject's overall perception of

the robot behavior in industry. On the other hand, in a laboratory research setting a human-subject that has not worked with an industrial robot felt safe and preferred working with the on-robot ToF-based dynamic SSM safety configuration. This would indicate that in an industrial setting, it is very likely an experienced human operator would have a similar or better experience. Thus, an on-robot sensing dynamic SSM setup should be investigated and verified in an industrial setting.

The learning bias of the human subject while performing the task will always be present, even though the human subject is asked to practice the assembly prior to the experiments. The robot performs the task based on a defined rule-based behavior, but the human subject's efficiency would likely get faster with the number of iterations of the assembly task. In this experiment, as the human subject's assembly task is simple, it is assumed that following the baseline experiments, the human subject is proficient in the assembly task. Thus, experiments quantifying and accommodating the learning bias and the cognitive load of the assembly task should be done.

During the HRC experiments, human-physiological signals were also collected. In the next section, we present how the dynamic SSM setup can be extended by incorporating the information of the human's affective state (the physical and mental state.)

5.4 Extending the Dynamic SSM

The current implemented Cyber-Physical System is used for HRC experiments for dynamic SSM, is capable of representing and recording the robot's motion and human-physiological state concurrently. To record this information, the system provides interfaces to the robot, sensors such as cameras, motion capture system and also to the biological/bio-metric data acquisition devices. As all these devices work on different sampling rates, this system helps in synchronous data acquisition and representation of the human-robot collaboration. There are two main future objectives for collection of human-physiological signals during this research. The first is to generate a database of physiological responses and human-robot interactions during HRC tasks. The second objective is to use this database to investigate how human physiological responses can be used to affect a robot's motion, and positively impacting the automation process. In other words, to build a 'physiological computing

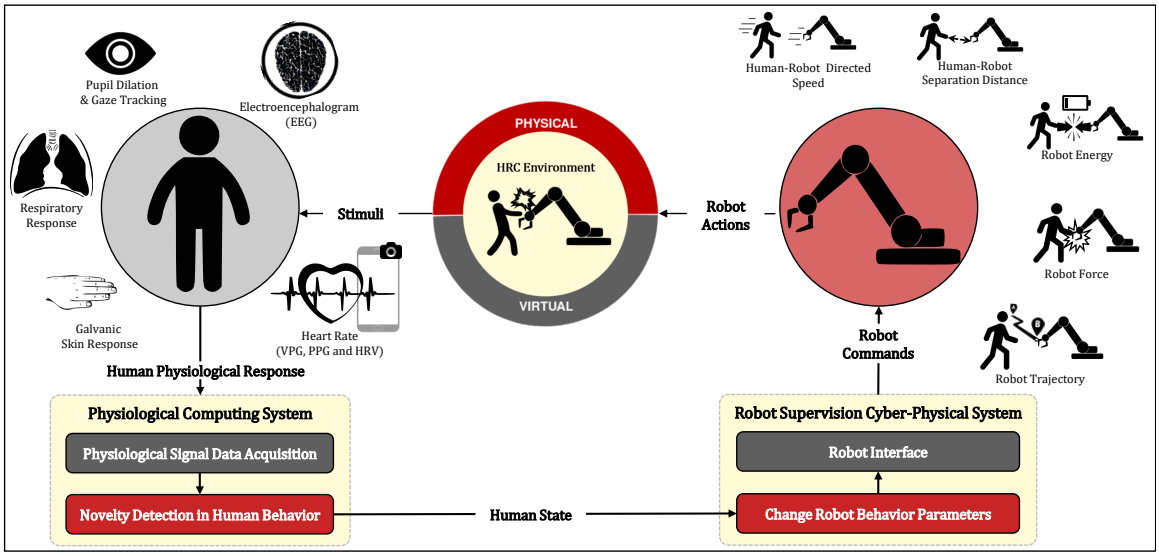


Figure 5.19: A Physiological Computing schematic diagram showing human-in-the-loop system.Video Link

system' [2] for human-robot collaboration in industrial and manufacturing environment [3, 18, 93]. A block-diagram of a physiological system is shown in Figure 5.19.

One of the main challenges in classification of human-physiological signals into actionable information is the labeling of the data. The implemented system is able to generate event-markers and the HRC state in real-time, which can be used to generate a labeled database. This can be used to train a classifier to infer information, that can be used to affect the robot action. An example scenario on how the robot behavior would change based on the physiological computing system for HRC can be considered: 'Two operators, A and B, are working with a robot in an industrial setting. Operator A has worked longer than B, and is comfortable working with the robot, as he/she can predict the robot behavior from experience. Operator B is new, and is a bit skeptical of the robot motion. The robot behavior changes in terms of the speed at which it moves, and how much distance it maintains when it is near the operator. This change is dependent on the operator's physiological state and behavioral patterns. In this scenario, for a better human-robot interaction the robot can move at higher speeds working near operator A, and move slowly working with operator B. As operator B gains more experience, the robot motion can adapt to it, thereby building

[2] "Physiological Computing is a term that is used to describe any technological system that incorporates physiological data from humans into its functionality or displays these data at the interface." - S.H. Fairclough [91]

trust and positively affecting the overall productivity.'

On accounting for the change in Human Affective State in Protective/Critical Safety Distance Calculation

The robot behavior in a dynamic SSM setup is dependent on three control variables. First is the directed speed v_o, which is a representation of how far the robot should start stopping or slowing down from the approaching human. The second variable that can affect the human is the cushioning constant, C_{dC}, which defines the minimum distance the robot should maintain from the human-operator. The third control variable is the acceleration/deceleration of the robot as it transitions between robot actions/operation modes. The transition between robot actions/operation modes is controlled using the online RML controller (refer Section 4.2.4.1). In the controller, the time-constrained parameter T_ψ can be set to control the amount of jerks or sudden changes in the robot motion, which can affect the human affective state during HRC.

Let $p_a \in [0, 1]$ be the probability of a detection of an anomaly/novelty in the human state that is indicative of his/her comfort level using psychophysiological signals. This probability can be incorporated in the calculation of the protective safety distance (PSD) as

$$d_C(t_0) \geq \underbrace{(1 + p_a k_{v_o}) v_o}_{\text{directed speed}} (T_R + T_{stop}) + v_R T_R + B + \underbrace{(1 + p_a k_C) C_{dc}}_{\text{cushioning constant}}$$

where k_{v_o} and k_C are scaling factors for the directed speed v_o and cushioning constant C, respectively, and are weighed by the probability p_a. In the above equation, the robot braking distance (B) is assumed as the maximum braking distance required by the robot to come to a complete stop as defined by the robot safety guidelines (refer to Section 4.2.1). This is a conservative estimation. In the implemented system, this is controlled using the RML controller to brake the robot with time-constraints such that the distance is always less than the maximum braking distance B.

In the next section, we summarize the results of the human-subject experiments and conclude.

5.5 Intermediate Conclusion

The dynamic SSM safety configurations are evaluated for an HRC setup where the human agent performs an assembly task with the robot. The experiments are performed with 19 human subjects with different SSM safety configurations comparing the SSM approaches TriSSM-Vo, TriSSM-Vr, and Tri-SM and the sensor modalities *Real*, *Ideal*, and *LiDAR*.

The human-subject experiments validated that the least productive dynamic TriSSM approaches using the *Real* (8-node ToF sensor array prototype) and the *Ideal*(motion capture and digital-twin) sensor modalities, are more productive than all the Tri-SSM approaches using 2D *LiDAR* sensor. It also proved with significant evidence that a dynamic TriSSM safety approach that considers the relative velocity of both human-agent/object (non-stationary) and the robot links (TriSSM-Vo) is safer, more anticipatory (predictable), and comfortable for the human agent to work with, than the safety approach that considers just the velocities of the robot links (TriSSM-Vr).

It also showed that the humans felt as safe and comfortable working with TriSSM safety configurations using *Real* ToF sensor arrays, as when using the motion capture and digital-twin based *Ideal* measurements. This was different than the initial claim and results in the previous Chapter 4, that uncertainty and error in distance measurements in *LiDAR*-based dynamic SSM would affect the perceived safety of the human-subject during the task. However, according to the objective evaluation, the safety metric reported *Ideal*-based SSM setup safer compared to *Real*. This result could be because of the size of a human as an obstacle, and the responsiveness of the ToF-based sensor arrays compensate for the uncertainty and minimum-distance error. Thus, resulting in a not so significant change in the perceived overall robot behavior by the human subject.

In addition to the dynamic SSM safety, the human-physiological signals were also recorded while performing the experiments for future use. A subsystem was integrated with the dynamic SSM setup to monitor and record the human physiological signals, and the human-robot collaboration state information synchronously and concurrently.

The results of the experiment corroborated and validated that *Real* and *Ideal*-based TriSSM-Vo is an optimal balance for safety and productivity and that ToF-based sensor arrays mounted at the centers of the robot link perform satisfactory in terms of safety. The performance of dynamic SSM was more efficient and preferred by the human subjects

compared to 2D LiDARS.

This concludes the research done for implementation and validation of dynamic SSM using on-robot ToF sensor arrays. In the next chapter 6, we present the hardware and software details of the 8-node ToF sensor array prototype and a 32-node prototype developed during this research. We also offer an analysis of quantifying the sensing volume (also known as detection coverage) of the ToF sensor arrays for varying placements on the robot links. The overall conclusion and future work are discussed following the next chapter in Chapter 7.

Chapter 6

Implementation of ToF Sensor Array

In this chapter, the design and implementation of an 8-node ToF sensor array prototype used in this research for the implementation of dynamic SSM is presented. The drawbacks due to lost coverage and placements of these sensors are discussed. To address some of these drawbacks, a second prototype with a modular design and up to 32 ToF sensor nodes is implemented, and sensor characterization tests in-terms of accuracy, overlap in coverage, and ranging under motion are performed. An analysis in terms of placement of the ToF sensor arrays to quantify the sensor volume coverage of the robot workspace is done using the simulated sensor setup in the digital-twin, and its results tabulated and discussed.

6.1 Prototype v1: An 8-Node ToF Sensor Array

This section presents the design and implementation of an intrinsic monitoring setup using three ToF sensor arrays with eight sensor nodes mounted in the center of the robot links. In Chapter 3, a test setup was used for testing the hardware and the performance of VL53L1X (single unit ToF lidar) based MappyDot sensor nodes. The placement of the ToF sensor arrays and its viability for implementation of dynamic SSM safety configuration was done in Chapter 4. The eight ToF sensor nodes are daisy changed to form a ToF sensor array (or ToF rings). Three ToF sensor arrays are placed at the center of the shoulder, elbow and tool links of the robot, as shown in Figure 6.1(a). A ToF sensor array consists of eight ToF sensor nodes, an Inertial Measurement Unit (IMU), and a central node, where a Teensy 3.2 micro-controller is used to interface with the sensors over I2C. An LED strip at the tool ToF sensor array is also implemented for visual feedback. This intrinsic sensing system for dynamic SSM was marked as Prototype v1.

Next, we present the hardware design details for the Prototype v1 ToF sensor array.

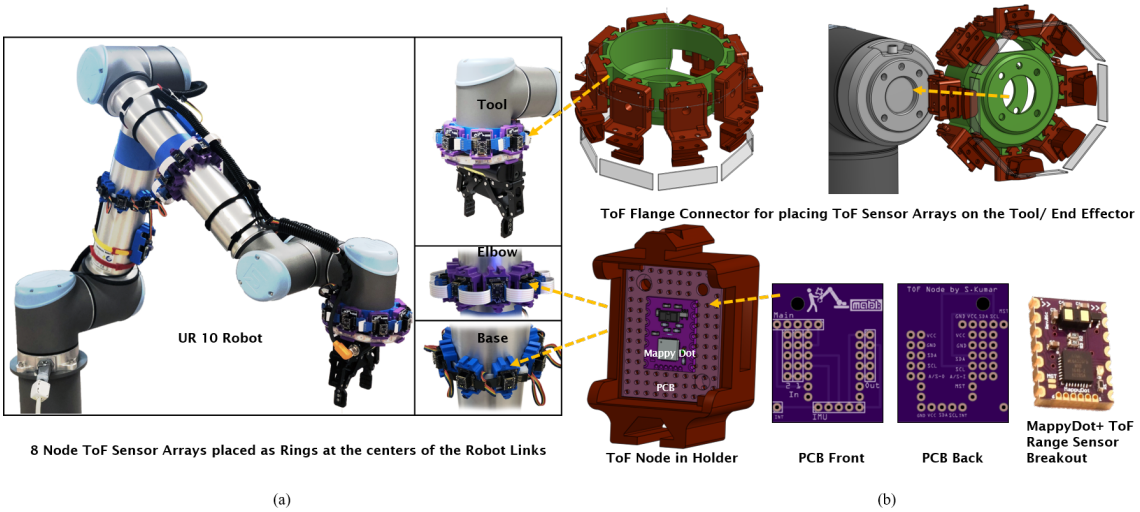


Figure 6.1: (a) The ToF sensor prototype v1 placed on the robot links - base, elbow, and shoulder. (b) The assembly of the ToF prototype used to mount the sensors on the robot links. A 3D printed tool flange and a fixture for a ToF sensor node. The PCB breakout used to interface and daisy-chain the MappyDot - VL53L1X ToF sensor module.

6.1.1 Hardware

The hardware implementation for placing the ToF sensor arrays on the robot links, required design of enclosures. These 3D models of these enclosures are shown in Figure 6.1(b). The links of the UR10 robot used in this research are cylindrical in shape. Hence, the enclosures designed for the base and the elbow ToF sensor array were similar to 'hose clamps' or a 'watchband' around the cylindrical links. To not restrict the robot's last (*wrist-3*) joint motion, and allow attachment and space for end-effector, a tool-flange fixture was 3D printed which rotated with the last (*wrist-3*) joint of the robot. Each sensor node was housed in an enclosure with a breakout connection board for the Mappy Dot ToF sensor node. These enclosures were clamped on the centers of the base and elbow links and the tool-flange fixture attached to the end-effector. The positions of the rings mounted on the robot were determined by the center of the link as specified by the UR10 robot's 3D CAD model.

The schematic of the components for the Prototype v1 ToF sensor array is shown in Figure 6.2. The ToF sensor nodes and IMU in the array were connected as slave devices on an I2C bus. A PCB breakout for the sensor nodes was designed for daisy-chaining

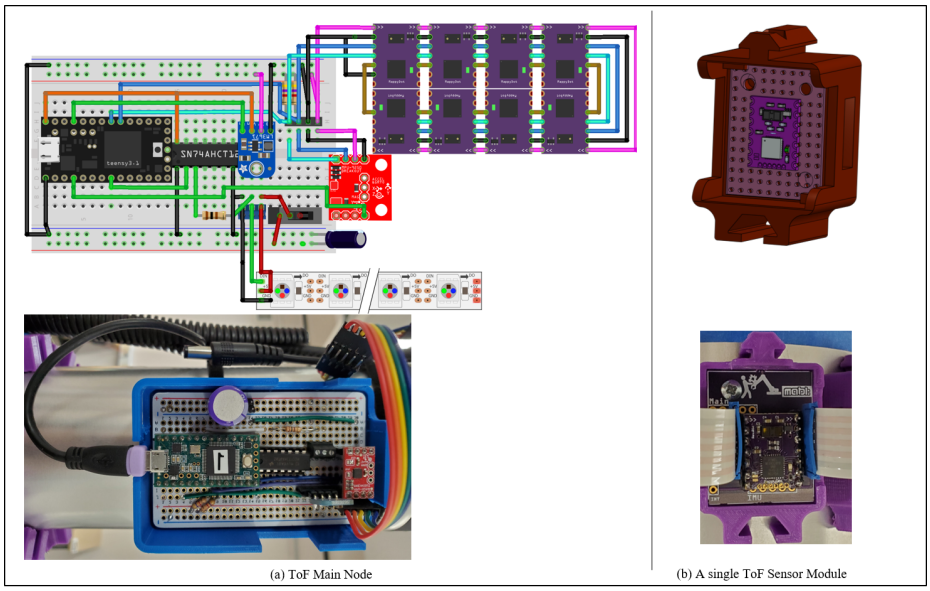


Figure 6.2: (a) The ToF Main node schematic for the Prototype v1 and its assembly in the setup. (b) A single ToF sensor module in the ToF sensor array with in the 3D printed ToF holder fixture.

the sensor nodes. The first node in the array was connected to the central micro-controller node, termed as the *ToF Main* node. The IMU was also connected to the I2C bus. A Teensy 3.2. micro-controller was used as the I2C master for collecting the data from the sensors.

In the next section, we present the software details for the Prototype v1 ToF sensor array.

6.1.2 Software

In a ToF sensor array, the digital communication protocol used is I2C. The MappyDot ToF sensor node contains the VL53L1X ToF sensor IC and an ATmega micro-controller (μC). Each ToF sensor node performs continuous measurements at $30Hz$ and maintains the measurements in a buffer for I2C master, the Teensy micro-controller (μC) on the main ToF node to read. The distance measurement data is smoothed using a low-pass filter on the MappyDot sensor node.

The data available from each ToF sensor node:

- **Distance measurement** of the object in the FOV of the sensor. This measurement is a 16bit unsigned integer value of the distance reported in *millimeters*. The range of

the sensor measurements lies between 30*mm* and 1500*mm*.

- **Standard Deviation** value reported is used as the confidence in the measurement reading. The closer the object detected to the center of the FOV axis, higher the confidence in the reading. This was used to filter low confidence reading.
- **Error Code** was either the status or an error reported by the VL53L1X ToF IC following each measurement. This was helpful in identifying, if a sensor node was not initialized, the distance measured was out of the detection range, the object was too close to the sensor, or the distance measurement was skewed due to interference with an external source of light of the same wavelength or from cross talk of other sensors. This information was important to monitor the status of the ToF sensor array.
- **Signal Strength** was the signal amplitude of the phased infrared laser pulse following a distance measurement. This value can be used to determine if the performance of sensors in changing light and detection of objects of variable reflectance (intensity of light reflected from the object) and sizes.

The Tennsy μC performs read operations on all I2C devices on the bus to generate a sequence of measurements for each sensor node and the IMU data. The first ToF node connected to the main node is used to inform the Teensy μC when new distance reading was available on any of the ToF sensor devices. The data from all sensor nodes and the IMU is placed in a delimited sequence of characters, and sent as serial packets over USB for processing. An RGB LED strip is also controlled using a one-wire interface. Each LED is addressable, and any RGB color can be set individually. The LEDs used are WS2812s IC. This is used as visual feedback of the distance measurement.

The setup of the connection of the Prototype v1 is shown in Figure 6.3. The three ToF sensor arrays (base, elbow, and tool) are connected over USB to the central computer. On the central computer, three independent threads of the program are used to decode the available data, pre-process it, and localize it with respect to the center of the robot link, as described in Section 4.1.2.3. The UR10 robot joint poses are determined using RTDE and ROS based communication, and the detected 3D points are transformed w.r.t. the robot static base frame. The detected 3D points are then visualized in RVIZ graphic environment.

Next, we discuss the implementation and testing of the Prototype v1.

6.1.3 Implementation and Setup

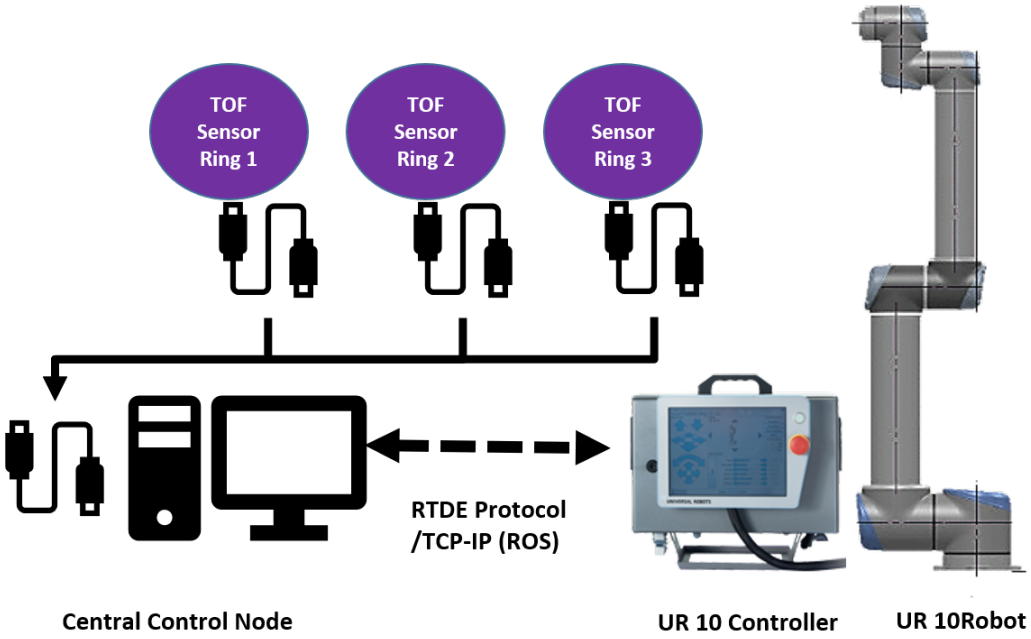


Figure 6.3: A block diagram showing the communication setup of the ToF sensor arrays placed on the robot links.

The implementation and testing were done in two phases. First, the tool ToF sensor array was assembled with the sensor nodes and the main node in a 3D printed enclosure. The software and hardware setup was tested and visualized. Second, all the ToF sensor arrays were placed on the centers of the robot links, and the complete intrinsic-sensing setup was tested.

The first phase examined the generation of 3D points with respect to the center of the ToF sensor array. The IMU orientation information was associated with the center of the array and visualized in RVIZ [107]. The colors of the LEDs nearest to a ToF sensor node were changed based on the distance measurement. The colors of the LEDs were determined as: a range of $[30mm, 500mm)$ RED, $[500mm, 800mm)$ BLUE, $[800mm, 1300mm)$ GREEN and out of measurement range the LEDs were off.

The centers of the links were carefully measured to coincide with the robot link centers

as reported by the URDF [1]. The tool-flange fixture was attached to the last joint of the robot, and the base and elbow enclosures were clamped to the robot links. The block diagram of the setup is shown in Figure 6.3. The robot information is processed using RTDE protocol and relayed to ROS. The robot 3D CAD model is visualized in RVIZ and updated in real-time based on the joint angles reported by the robot interface. The ToF sensor arrays are connected to a central computer over USB serial. The ToF sensor arrays are identified by associating the Teensy μC serial number also known as device id to the location of their placement on the robot links. The complete Prototype v1 setup is tested by visualizing the 3D points in real-time in RVIZ for all ToF sensor arrays when the robot is in motion.

In the next section we present the results of the tests.

6.1.4 Results

In this section, the results of the tests of Prototype V1 are presented. Figure 6.4 shows the visualization of the 3D points with respect to the center of the ToF sensor array, in RVIZ. This visualization and the data reported by the ToF sensor array was analyzed in the first phase of testing, to examine the hardware and software of a single ToF sensor array (placed at the tool link of the robot). The update rate of the distance measurements ranged from $27Hz$ to $30Hz$.

In the second phase of testing, three ToF sensor arrays were placed on the robot links. In this test, the information from the three ToF sensor arrays and the UR10 robot kinematics was combined and visualized. The sensor values were monitored in real-time and plotted in 3D with the robot in RVIZ. The result of the second phase of tests is shown in Figure 6.5. The update rate did not change, as each ToF sensor array was interfaced independently in mutually exclusive threads. The minimum distance measurement measured from all the ToF sensor arrays was plotted, as shown in Figure 6.6. The plot shows the calculated human-robot minimum distance; not filtered and moving average filtered data. The trend is compared to the ground truth minimum distance measured using the motion capture system setup. It was observed that the sparsity of the sensor nodes (eight nodes) resulted in blind-spots and abrupt jumps in the range reading(s). The measured response

[1] The Universal Robotic Description Format (URDF) is an XML file format used in ROS to describe all elements of a robot.

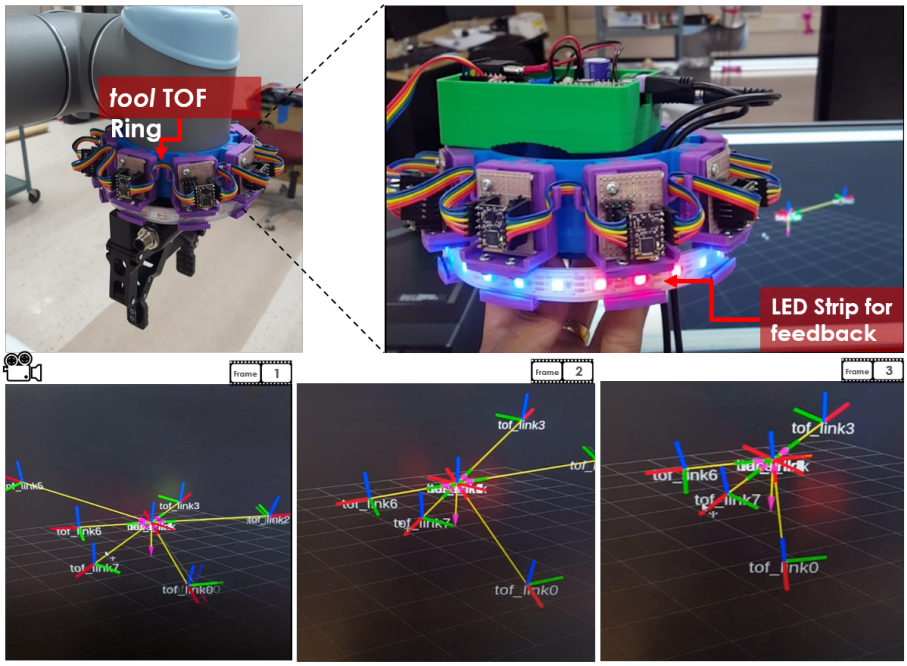


Figure 6.4: The 8-node ToF sensor array Prototype v1 sensor values monitored in real-time and plotted in 3D using RVIZ. This setup was used to verify the working of a ToF sensor array. Video Link https://youtu.be/DSPsAcuQduQ

when coverage is lost, resulted in spikes representing out-of-range or large distance measurements. The uncertainty and spikes in the data sometimes were also due to erroneous readings. The raw distances (not-filtered) and the filtered (low-pass) distances were compared to the ground truth. A filtered distance resulted in a shift in the report of the sensor distances and attenuated the response. This was not desirable as SSM is time-constrained, and could adversely affect the behavior and the reaction-time of the robot. Hence, filtering was not used during the implementation of dynamic SSM (refer Section 4.2.6).

In the next section, we discuss the limitations of the 8-node ToF sensor array prototype.

6.1.5 Limitations

The performance of the Prototype v1 was satisfactory in terms of distance accuracy for the range of 0.03m (30mm) to 1.3m (1300mm). The measurements were robust and were

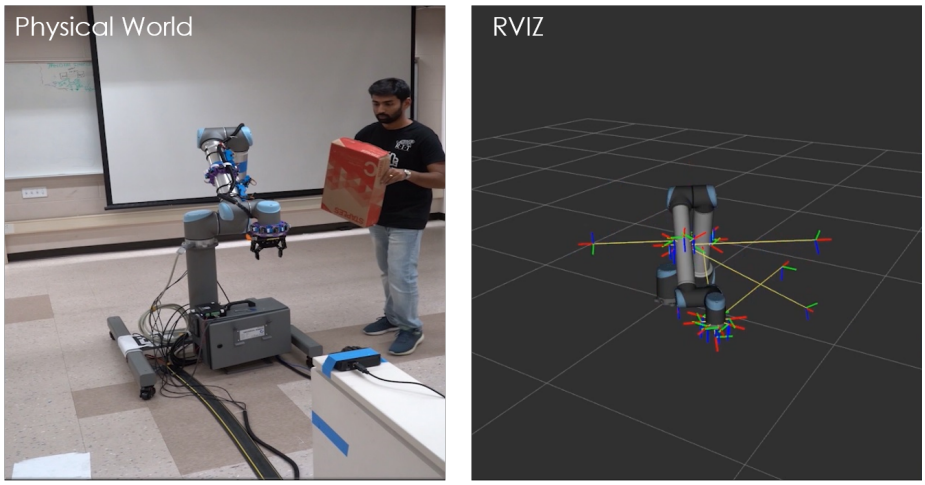


Figure 6.5: The implementation of the Prototype v1 placed on the robot and its sensor values monitored in real-time and plotted in 3D using RVIZ. Video Link https://youtu.be/mQw0rl0upms

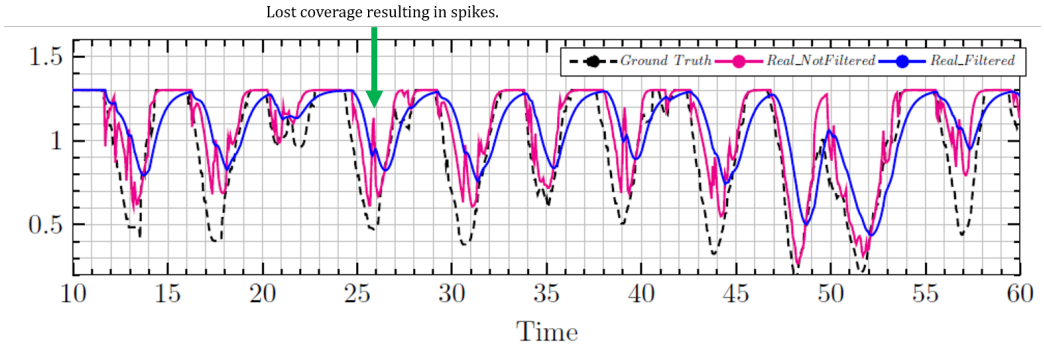


Figure 6.6: A snippet of the minimum distance measurement for all ToF sensor arrays is plotted. The trend is compared to the ground truth minimum distance measured using the motion capture system setup.

reported in real-time at $27Hz$ to $30Hz$. This distance information can be used for implementation of dynamic SSM. The minimum distance of the obstacles in the surroundings can be directly calculated from the robot reference frame with minimal computation and information.

A MappyDot ToF sensor node is a third-party package of the VL53L1X ToF sensor chipset. It has a micro-controller associated with each sensor node. The computation capabilities of the micro-controller are wasted, and the overall consumption of power is higher. In Prototype v1, there are in total nine μC used for interfacing with the ToF sensor

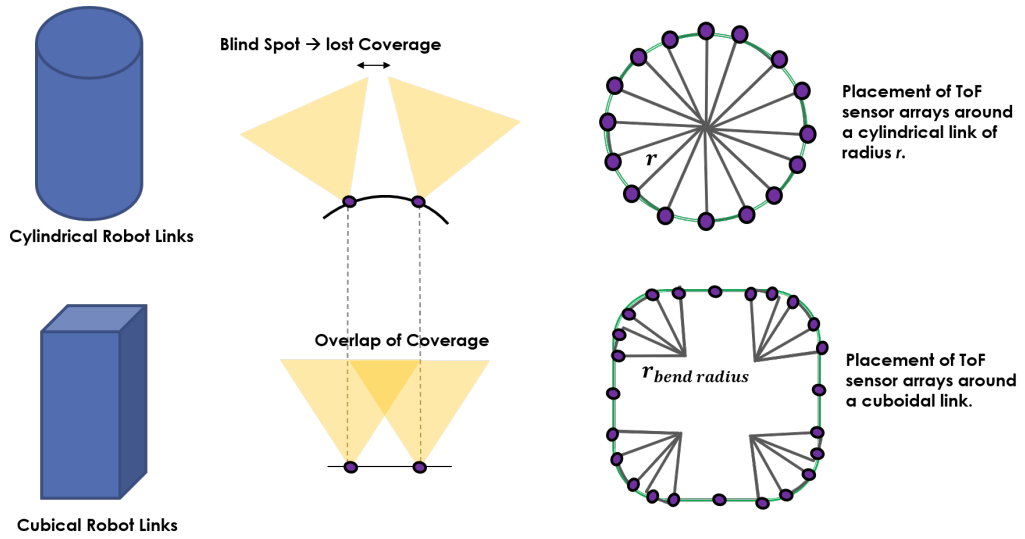


Figure 6.7: A schematic of the lost coverage resulting in blind-spots for cylindrical links. The placements of the sensors around the cylindrical and cuboid links.

arrays. This computation can be done using a single μC, such as the Teensy 3.x. Moreover, the cost of each MappyDot ToF node is higher in comparison to a breakout of the VL53L1X chipset. The overall footprint of the sensor array can also be decreased.

The robot links can be of varying shapes, cylindrical or cuboidal. As shown in Figure 6.7, the density of the sensor nodes in an array is critical to avoid lost coverage due to blind spots. The density of the arrays would vary based on the size of the robot links and the footprint of each sensor node. Hence, a smaller footprint of the sensor node and modular expandable design of the ToF sensor array is needed. This setup should also be adaptable to the multiple shapes and sizes of the robot links. In this research, the UR10 has cylindrical links of varying diameter, and the ToF sensor array has eight sensor nodes for all links. The resulting human-robot minimum distance is plotted and shown in Figure 6.6. The measured response when coverage is lost results in spikes representing out-of-range out-of-range or large distance measurements.

The three ToF sensor arrays are connected over USB serial to the central computer. There are three separate USB connections to each ToF sensor array main node connected to the Teensy μC. For improving cable management and reducing clutter a CAN bus daisy-chained implementation can be used. CAN is an industrial standard digital communication protocol and is used for robust and hi-speed data transfer.

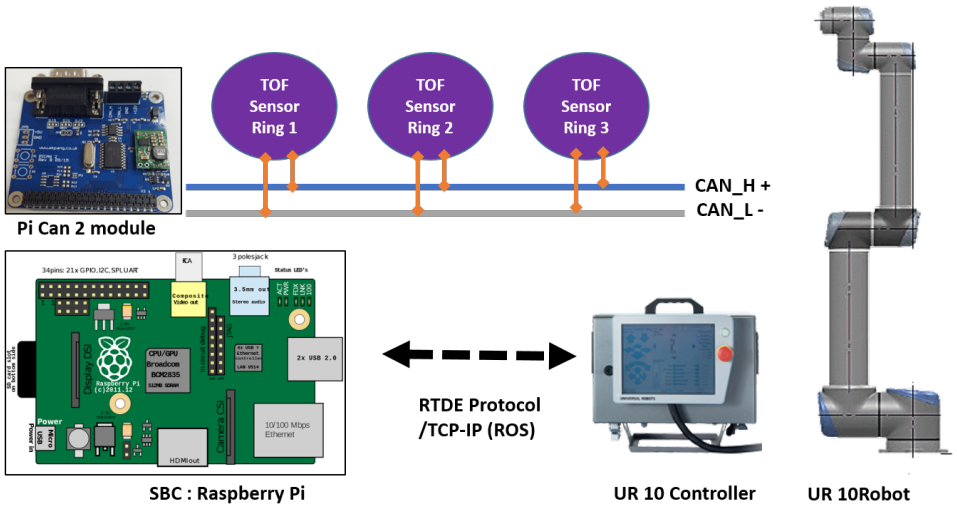
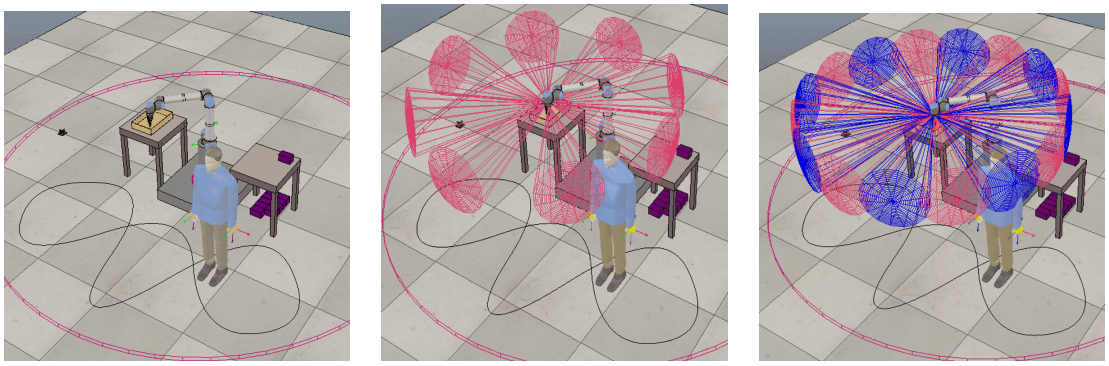


Figure 6.8: The setup of daisy-chained ToF sensor arrays and interface of the robot controlled using CAN bus and a Raspberry Pi single board controller.

ROS is used as the communication layer for the aggregation of the data from the robot and the ToF sensor arrays. ROS can cause loss of data as it discards packets of data not used. It is also not compliant with industrial standards. Hence, other communication layers such as ZMQ or ROS 2.0, an industrial compliant version of ROS, should be used. In this setup for testing with RVIZ and faster prototyping, ROS and ZMQ were used.

The central computer can be replaced with a single board computer (SBC) such as a Raspberry Pi for an embedded implementation of this setup. This is feasible due to the low computation requirements of calculating the minimum distance. The SBC can be used for the generation of the 3D points detected by the ToF sensor setup in reference to the robot base frame. A schematic of the setup is shown in Figure 6.8.

In this research, Prototype V1 was used for implementation of the intrinsic sensing SSM setup. In conjunction to this implementation, in order to address some of the drawbacks, especially the lost coverage, a research for designing a second prototype was undertaken. The details of this work are described in the following Section.



(a) (b) (c)

Figure 6.9: (a) A setup with Universal Robot UR10 and a human sharing the workspace. For reference. (b) An 8-node ToF sensor array mounted on the end-effector of the UR10 robot.(c) The proposed 16-Node ToF sensor array mounted on the end-effector of the UR10 robot.

6.2 Prototype v2: A 32-Node ToF Sensor Array

This section presents the design and implementation of a modular ToF sensor array where a maximum of 32 ToF sensor nodes can be placed on a robot link. This second version of the prototype (Prototype v2) is designed to be a Minimum Viable Product (MVP) [2] that can be used for testing purposes as an electronic safety device for HRC setups in industry. This prototype addresses the challenges of modular design, small footprint, increased density of nodes, and direct communication with all the sensor nodes using a single micro-controller. The Figure 6.9 shows a comparison of the *tool* ToF sensor array of Prototype v1 and v2 in terms of sensor nodes density per array in simulation. The ToF sensor arrays/rings (*base*) of Prototype v2 and v1 are shown placed on the shoulder robot link in Figure 6.10.

This sensor setup was developed in parallel to the implementation of dynamic SSM implementation using Prototype v1. Detailed design and development of this sensor setup is presented in [4, 28] and summarized in the following sections. Multiple tests were performed to measure the performance of the setup. The tests performed were single sensor characterizations, the sensor coverage overlap characterization, and the sensor ranging performance under motion.

[2] A minimum viable product, or MVP, is a product with enough features to attract early-adopter customers and validate a product idea early in the product development cycle.

Figure 6.10: Sensor density comparison between Prototype v1 ToF Ring (bottom ring array) and Prototype v2 ToF Ring (top ring array).

Next, we highlight the hardware details of the second prototype.

6.2.1 Hardware

The ToF sensor array was comprised of three nodes, the master μC node, the ToF sensor nodes, and expander nodes. A hardware block diagram is shown in Figure 6.11. The master node has full control over all expander and sensor nodes on the I2C bus of the given master.

As per the hardware constraints of the IO Expander ICs on the expander nodes, a maximum of eight expander nodes with 32 sensors could be attached to a single I2C bus at one time. With a sensor spacing of 12mm (each sensor node right next to each other), the ring array would have a circumference/length of 384mm or 1.512ft. The maximum length based on the connections of the FPC flexible cable [3] can be 960mm for 30mm spacing, and 1600mm for 50mm spacing. A schematic showing this modular design is shown in Figure 6.12.

[3] Flexible Printed Circuit (FPC) cable or a Flat Flexible Cable (FFC) are flexible cables for interconnections between printed circuit boards (PCBs).

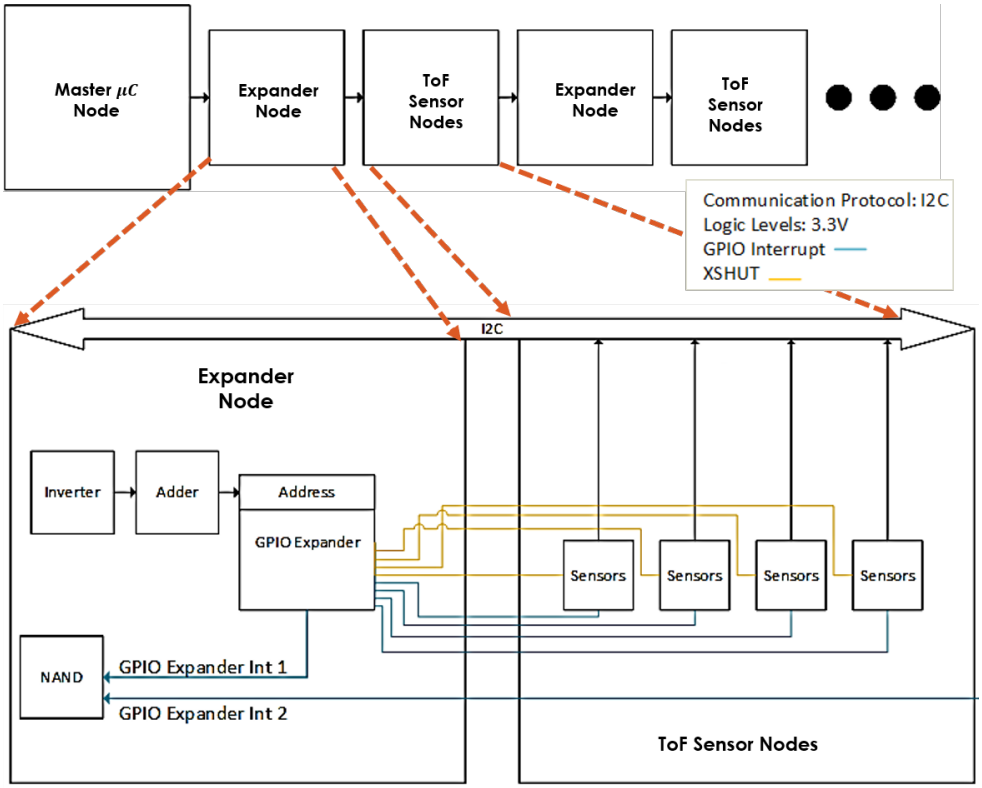


Figure 6.11: A hardware block diagram for the ToF Expander and Sensor node for Prototype v2 [28].

6.2.1.1 Master Micro-controller Node

The master node design consists of a micro-controller, a CAN transceiver for future system monitoring, and a power regulator. The current design uses a Teensy 3.2 as the master micro-controller. The Teensy controls the I2C bus and all slave devices on the bus. It also controls the master address pin to define the first IO expander in the sequence of expander nodes. Additionally, the main node receives the sensor interrupt line, which can fire an interrupt per distance data packet from a sensor node. Currently, the micro-controller outputs its data via serial USB communication. This configuration was chosen to match the first prototype for easier integration with the existing dynamic SSM system, which uses serial data as an input to the SSM control loop.

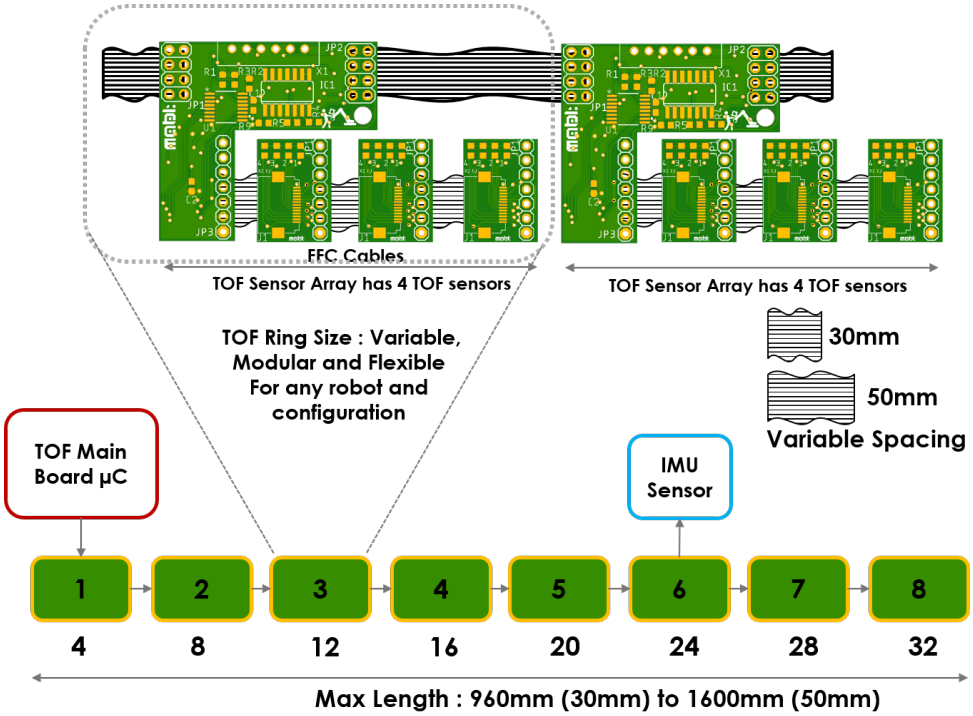


Figure 6.12: An assembly diagram showing the modular design of Prototype v2 sensor and the daisy-chaining of 32 sensor nodes.

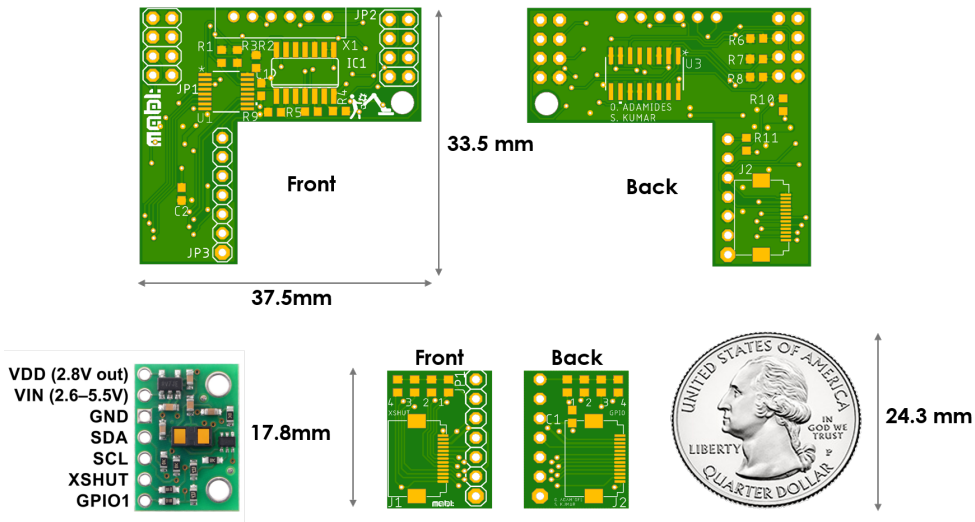


Figure 6.13: The PCB size comparisons of the ToF Expander nodes, the ToF sensor nodes and the Pololu VL53L1X breakout board.

6.2.1.2 Expander and Sensor Breakout Nodes

The schematic and dimensions of the expander node PCB, the sensor breakout node PCP, and the Pololu VL53L1X sensor breakout module are shown in Figure 6.13.

The expander node consists of a TCA9534A I2C IO expander, CD4011BNSR NAND gate package, and a CD74HC283M96 4 Bit Binary Full Adder. The design decision of using IO expanders stemmed from the STMicroelectronics application note for chaining multiple VL53L1X sensors together [26]. All reads and writes to and from the IO expander are performed over the I2C bus. The NAND gates and and a full adder increment the address for each expander node. Each IO expander address is controlled by three hardware pins. Therefore, the full adder increments through the address pins by one each time a new expander is added into the chain. Expander 1 is set to 000 through a NAND gate and an IO pin on the master micro-controller. Two NAND gates are combined to make an AND gate. This gate is used to transfer incoming interrupt signals from other expanders to the master node. The choice of an AND gate was made because the interrupts from the IO expanders were active low. Therefore, if any expander or combination of expander nodes pulled the line low, the output line to the master interrupt input pin was pulled low and an interrupt was fired on the master micro-controller, indicating the availability of a new sensor reading on a sensor node. The master controller can check the status of the IO expander nodes to read the sensor node with the updated reading. This allowed the asynchronous and fast update of distance measurement data for the ToF sensor array.

6.2.1.3 Enclosures and Assembly

In Figure 6.14, the assembly of the expander node and sensor nodes are shown. The ToF sensor array enclosures are designed to maintain system modularity at the mechanical level. Without maintaining physical modularity, the benefits of the hardware modularity would be significantly minimized. The enclosure design needed to tackle three main structural challenges; enclosing the sensor breakouts, enclosing the expander nodes, and connecting these pieces together. The 3D CAD models of the enclosures and the assembled ToF sensor array mounted on the robot link is shown in Figure 6.14.

Sensor breakout nodes are attached to the expander node via *0.5 mm* pitch FPC cables. The cables are chosen for their small size and physical flexibility. The Pololu ToF board

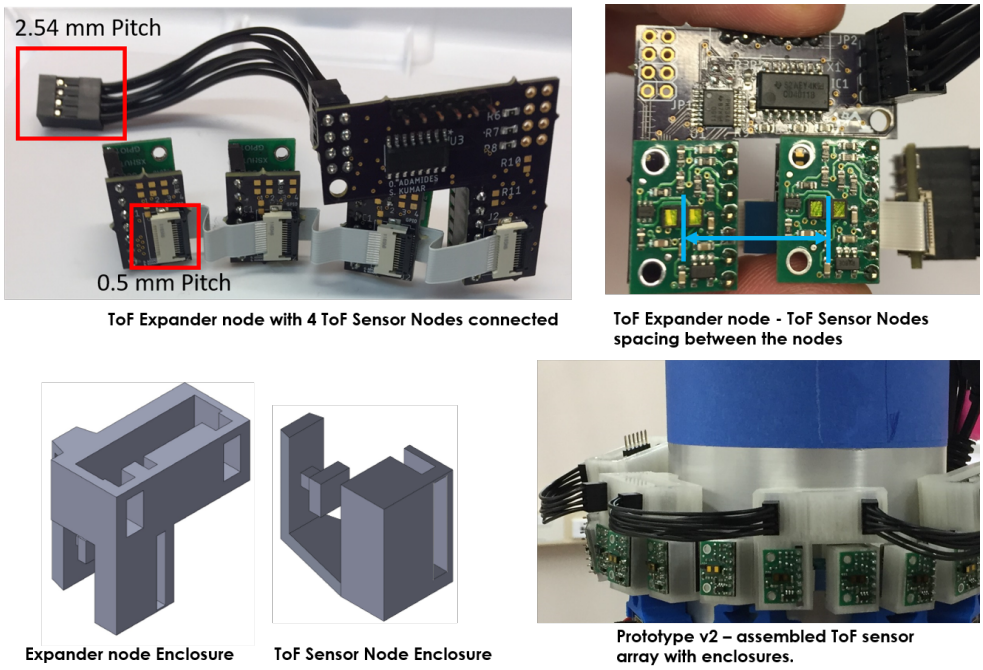


Figure 6.14: The Prototype v2 with the expander and sensor nodes in enclosure.

with peripheral circuitry is connected on top of the sensor breakout via a *0.1in* female headers, as shown in Figure 6.14.

6.2.1.4 I2C Peripherals

Prototype v1 included the ability to use an IMU for the sensor array orientation. In the original design, this was located on the main controller's breadboard. However, in this design, the IMU can be added as an additional peripheral I2C device. For example, the MPU9250 could plug into the same location on an expander board, where the main control board would connect (see Figure 6.12). With this setup, the IMU node can receive power, and connect to the I2C bus without adding a custom connection on the main board or an additional port on the expander boards. The implications of this configuration allow any type of I2C peripheral to be attached to this port. This setup could be extended out to I2C based LED drivers to generate visual distance feedback to the human in the robot workspace. Other environment monitoring peripherals that can be connected are temperature sensors, humidity sensors, haptic feedback, or acoustic transducers.

The software for interfacing with Prototype v2 is detailed in the next section.

6.2.2 Software

The data acquisition software on the main node is written for a Teensy 3.2 micro-controller (72 MHz Cortex-M4). The API libraries for interfacing and control of the VL53L1X ToF sensor node and the IO expanders are used to develop a wrapper library for interfacing with the ToF sensor array. The software is designed for two main functionalities, sensor initialization and data acquisition [28]. Each functionality tackles challenges posed by the hardware, as well as the modularity constraints of the setup.

The data acquisition is designed for maximizing overall speed of data acquisition. The sensor array's largest frequency bottleneck for acquisition is the ToF sensors themselves. A ToF sensor node provides a distance measurement at $30Hz$ for *4 meters*, and $50Hz$ for $1meter$. Therefore, the full ToF sensor array acquisition sequence is designed to fit within a $50Hz$ window. there are two approaches used for data acquisition using the Teensy; a purely interrupt-based approach, and a timer interrupt-based method to synchronize data acquisition. In the first approach, at the trigger of an interrupt signal, the master node checks all the sensor nodes with new information and performs read operation. The second approach uses timers to regulate sensor data acquisition and serial monitor transmission. Each timer fires an interrupt at a set period. On that period cycle, the timer interrupts the main loop routine to check the sensor nodes for new measurements. One timer is dedicated to reading the pin states of the IO expanders to determine the sensor addresses where new measurements are available, and the second timer is used to perform read operations.

The data is recorded in the same format as the Prototype v1, and is reported over USB serial to ROS and MATLAB. The data is not filtered on board. To test the speed and data acquisition, the data is reported and sent serially only when all the sensor nodes have a new sensor reading. This approach provides the lower bound of the data acquisition speed. This is not the approach used with Prototype v1.

In the next section, we present the implementation of the setup used for sensor characterization and testing of Prototype v2.

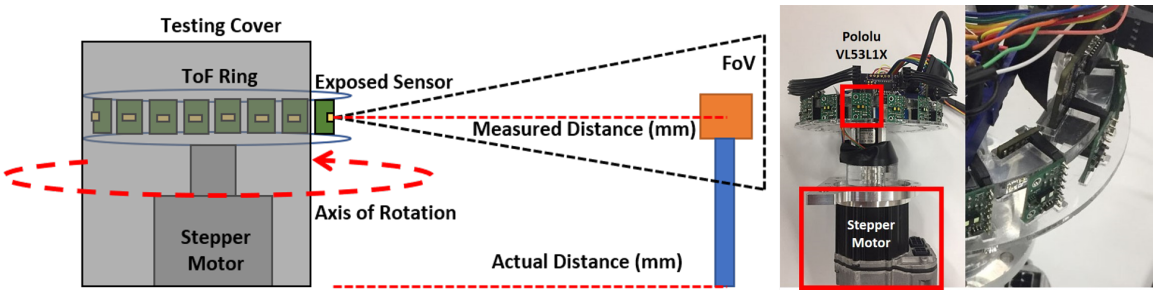


Figure 6.15: Schematic of the sensor characterization fixture. The prototype v2 with sixteen ToF nodes is outfitted on a stepper motor for testing ranging under motion [4].

6.2.3 Implementation and Setup

There are two Prototype v2 ToF sensor setups used for testing the performance. A full 32 ToF node array is used to test the load and data acquisition performance over I2C, and a 16-node ToF sensor array for sensor characterization.

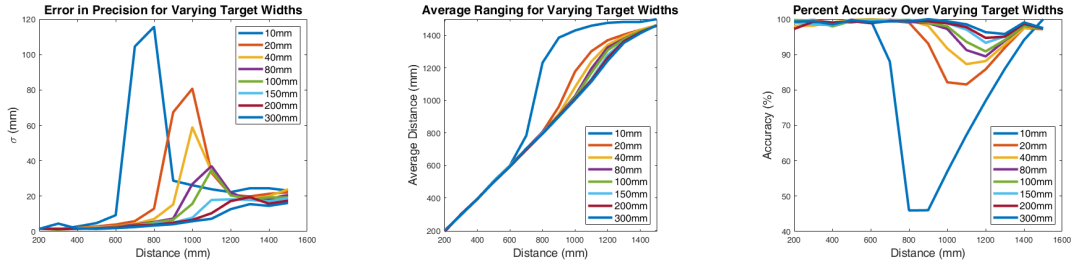
The prototype v2 ToF sensor array/ring is built to encompass a $4in$ ($0.106m$) diameter end-effector link. The link size is chosen to emulate the final link of UR10 robot. To minimize blind spots and optimize the ring's FOV, a sixteen ToF sensor node array, compared to eight in Prototype v1 is used during sensor characterization (see Figure 6.9).

A schematic for the sensor characterization test-fixtures is shown in Figure 6.15. A stepper motor is mounted to the bottom of the ToF sensor array test fixture. This is used to precisely rotate the fixture at a constant revolutions-per-minute (RPM) for testing. This fixture is implemented for testing the performance of measuring distance under motion. Around the test fixture, a cylindrical cover/guard is used with a viewing window. This guard ensures that at any given time, only one sensor is exposed for measuring the distance of the object per rotation.

Next, we present the results of the sensor characterization tests and the data acquisition performance of the second prototype.

6.2.4 Results

In this section, the results of the sensor characterization of the 16 ToF node sensor array and the data acquisition for the 32 ToF sensor array are presented and discussed. The tests for sensor characterization of Prototype v2 were single sensor characterizations, the



(a) (b) (c)

Figure 6.16: (a) Precision of sensor in mm measuring different object widths at different distances. (b) Measured distance vs. actual distance of targets with different widths. Error accumulates the smaller the object is as well as how much of the FOV it occupies. (c) Percentage accuracy of sensor ranging targets of different widths at different distances.

sensor coverage overlap characterization, and the sensor ranging performance under motion [4, 28]. The results of these tests are shown in the following sections.

6.2.4.1 Single Sensor Characterization

The first tests performed evaluated the VL53L1X ToF sensor node in a motionless application. These sensors nodes are the fundamental building blocks of the ToF array and validation of its effectiveness as a sensor provided insight into what the data represented. The single sensor characterization consisted of collecting ranging data for the detection of targets in several different scenarios. The ranging was performed for target of varying widths (ranging from $1cm$ to $30cm$) placed distances from $20cm$ to $150cm$ at $10cm$ increments.

Data samples were collected in sets of 500 points. These collections were averaged, and the standard deviation was taken at each distance for given target widths. The results showed decrease in the precision of the ToF sensor, as the target was moved away from the sensor. For any given distance, however, the object's width had a greater effect on this precision.

It can be inferred, that the occupied FOV has a direct correlation to the precision and accuracy of the sample [30] (as observed in Figure 6.16(a) and Figure 6.16(c)). If multiple objects were detected within the sensor's FOV, the distance output was the average distance among the objects. In the tests with different object sizes, the closer those objects were to the sensor relative to the background wall, the higher the accuracy in the estimate of the target's distance. This was because closer objects took up a higher percentage of the FOV

and the reflected signals were stronger, causing the estimated distance to fall close to the target. As objects moved further away, both metrics decreased. This decreased the accuracy of the object. There was a point at which the object could no longer be differentiated against its background. This was the point of the highest precision error. After this point, it was uncertain what the sensor reading represented, however, its constant nature represented the background it saw (as observed in Figure 6.16(b)).

These observations showed that the ToF sensor generated very reliable target acquisition out to 800*mm* for smaller targets (width between 20*mm* to 100*mm*), and out to 1000*mm* for larger 300*mm* targets. The 300*mm* target size is comparable to the average size of human body parts, including the head, waist depth and shoulder depth [85, 123].

6.2.4.2 Sensor Overlap Characterization

For dynamic SSM, higher accuracy in the calculation of minimum distance would result in enhancing the human safety [46, 85]. Sensor ranging overlap is crucial for the accuracy and robustness of the ToF sensor array setup. In Prototype v1, the blind spots resulted in some loss of coverage. In order to eliminate blind spots, the ToF sensor array, denser arrangement of sensor nodes was required.

A sensor node placed on a cylindrical robot joint detects objects in its FOV. By increasing the number of sensor nodes per array increases the sensing/detection coverage volume, but poses issues of portability, power consumption, and physical limitations. Tightly packed sensors on an arc of the cylinder, the FOV of these sensors would overlap. For a a robust sensor setup, a balance between the amount of FOV overlap and number of sensors in the array is desired. The relation of the desired amount of overlap of the coverage for a given robot link, and the spacing between the sensor nodes is modelled as shown in Figure 6.17. The equations showing the relation are:

$$r + d = r \cdot cos\left(\frac{\theta}{2}\right) + tan(\gamma) \cdot r \cdot sin\left(\frac{\theta}{2}\right) \tag{6.1a}$$

$$\theta * r = distance\ between\ ToF\ Nodes \tag{6.1b}$$

The goal of the Prototype v2 was to find the least amount of sensors that were able to

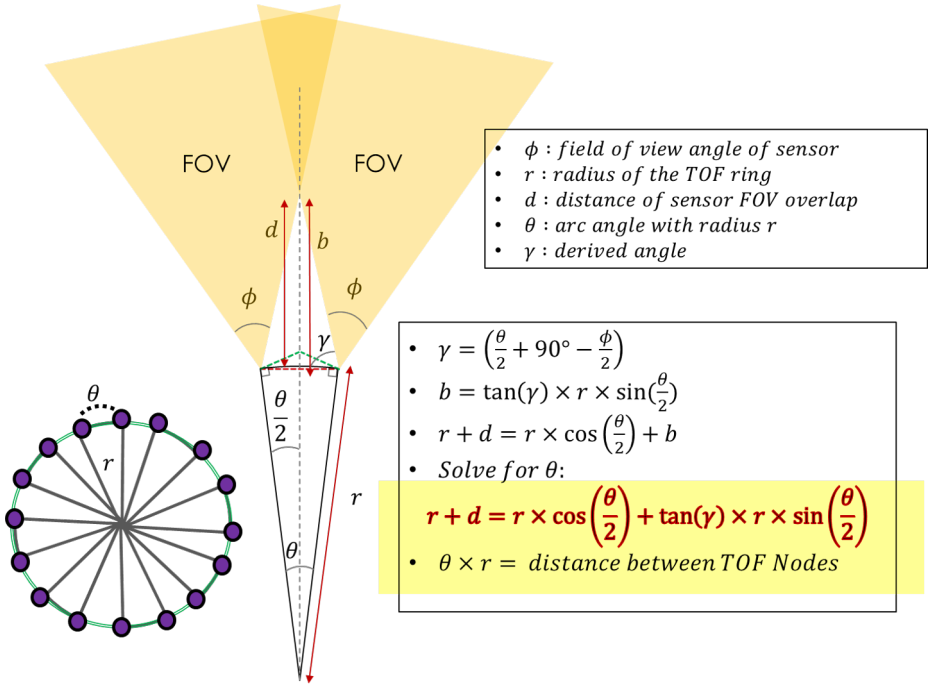


Figure 6.17: Geometric Model of two sensor overlapping at distance d on the circumference of a ToF array ring with radius r

achieve near-total coverage in FOV, as well as understanding how placement and positioning of the sensors affected this coverage (Figure 6.7 and Figure 6.17).

The distance of the sensors FOV overlap (b) was determined based on the minimum stopping distance of the UR10 robot [11, 59]. In this research, this was taken as 500mm. The sensing coverage overlap of the FOV was to start from this distance for complete coverage for greater than $b = 500mm$. In Eq. (6.1a), Eq. (6.1b), the desired spacing was calculated to fit sixteen sensors around UR10's robot link with radius r. Each of the sixteen sensors around this link were angled at 22.5° of the arc of the cylindrical link (θ). This information was used to determine the sensor spacing at an overlap distance of 517.45mm (also refer to Figure 6.17). This overlap distance was within 20mm of the desired stopping distance.

For validation, a sensor model was generated in MATLAB using Eq. (6.1a) and Eq.(6.1b). The results of this model are shown in Figure 6.18. A simple test was devised to validate the overlap of the sensor ring. A 20cm wide target was placed between two sensors and was positioned at interval locations away from the sensors. This particular size was chosen

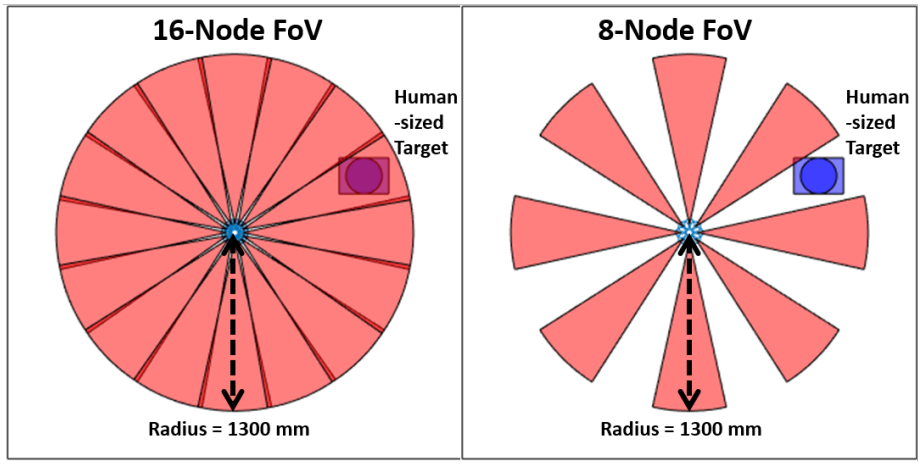


Figure 6.18: Comparing the field of view of an 8-node to a 16-node sensor array. The range is set to *1.2m*, the limit set for the tests. This shows how there are clearly visible blind spots in the 8-node array vs. 16-node array.

as it is, on average, the width of a human head [51, 123]. Overlap was seen at all tested distances from 300 mm to 600 mm. The target was seen below the calculated overlap distance target's width occupied at least one FOV at a time. It was observed that as the target was moved further away, the ranging deviation increased.

The distance error was calculated by taking the difference between the readings of the two sensors and verifying the overlap condition. The increase in error occurred because the target distance increased and the strength of the signal reflected back to the Single Photon Avalanche Diode (SPAD) matrix decreased [26]. These results match the behaviors seen in the single sensor test. As the target distance increased, the distance error increased. However, it can be observed that around the overlap location of 450*mm*, the error dipped before continuing on its increasing trend. A potential cause of this behavior could be attributed to the overlap region being the location where both sensors receive an equal amount of ranging data. Since they received a similar return signal, this could potentially explain why the difference between the readings became slightly less (as seen in Figure 6.19).

6.2.4.3 Sensor Ranging Under Motion

This test performed on the 16-node ToF sensor array was for testing the accuracy of distance measurements under motion. Here, the test fixture was used to emulate the movement of a ToF sensor array attached to a robot arm. The test case represented a 1-DOF arm rotating at

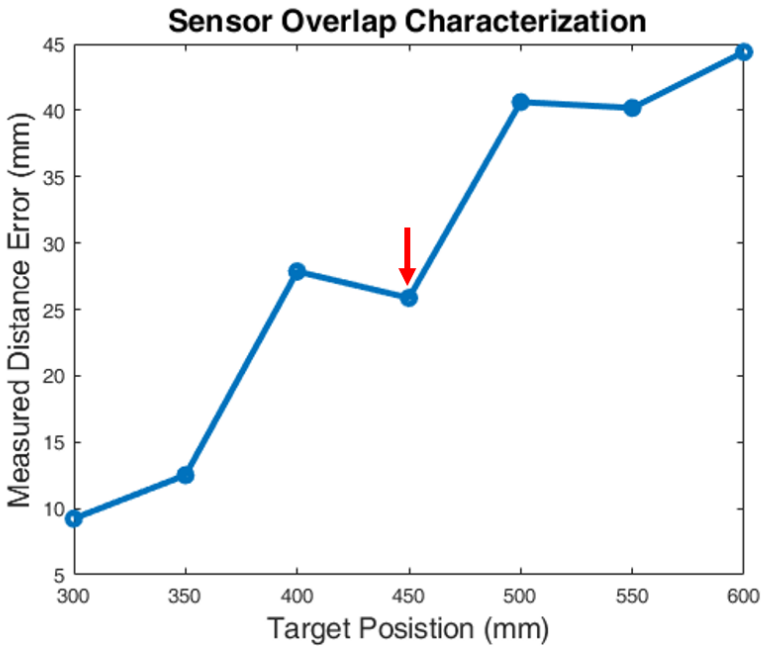
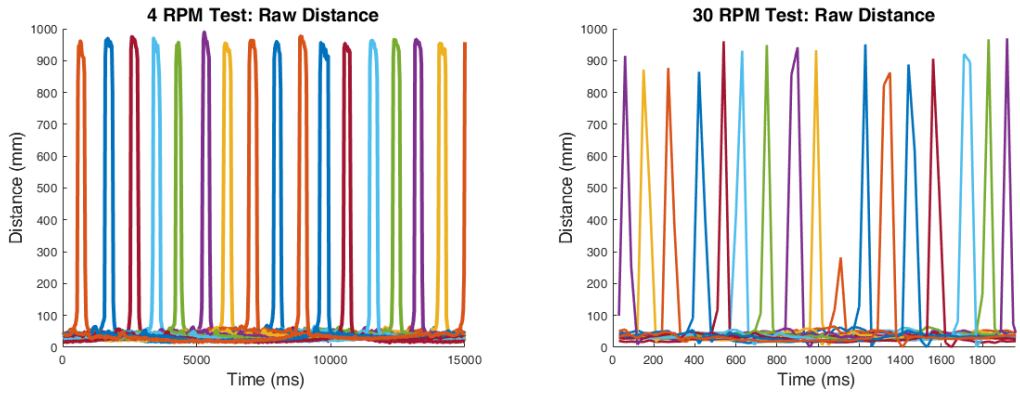


Figure 6.19: Distance error (the absolute difference in the distance measurements of the overlapping sensor) when measuring a 20*cm* object at different distances away from the sensors. The overlap is tested between 300*mm* to 600*mm* for a 20*cm* wide object. The overlapping region in this setup is seen at 450*mm* where the error dipped.

constant RPM. The sensor ring test fixture was built with a stepper motor as its base (Figure 6.15). The sensor ring array was surrounded by a target cover/guard with a viewing window that allowed only one sensor to collect data at a time. This guard made data interpretation much easier by clipping the data from other sensor nodes and also keeping the ambient light interference low. Using the guard, the measurements of all non-target data were less than 100*mm*.

The motor was rotated at varying RPMs for varying target widths. Each target width was tested from 4*RPM* (24°/*s*) to 60*RPM* (360°/*s*). This test range was determined based on the slowest rate that allowed each sensor to detect the object in a 15 *second* test duration. The fastest RPM was chosen to test the sensor-system significantly above the operating joint-velocity of an industrial robot.

Each set of RPM captures, for a set target width, were bench-marked using precision and accuracy evaluation. This was similar to the approach taken in the static sensor tests. The standard deviation (σ) for a given target width at an average distance μ was used to determine the valid target capture range. The capture range $\delta = \mu + 5\sigma$. Any points outside



(a) (b)

Figure 6.20: The RPM captures for a 8cm wide target at 1m distance away. Each color peak represents a sensor identifying the target through the target hole. (a) At 4RPM, consistent detection of targets. (b) At 30 RPM, sensors began to miss the target.

of this range were considered a failed target acquisition. This acquisition representation was used to determine which amplitudes were valid data points to perform signal analysis on. The standard deviation was taken across all valid points for each speed setting. The average value was taken for each valid pulse in an RPM set. These valid pulse averages were averaged together. MATLAB was used for the analysis of the data.

The raw data generated a periodic square wave across the sixteen sensors. The 4 RPM case in Figure 6.20(a) showed a clear visual representation of the ToF sensor array behavior. The height of each pulse was the distance at which the sensor saw the target. In stable operation, the period and amplitude of the square wave remained constant. The sensor system operation was stable from $4RPM(24°/s)$ up to $20RPM(120°/s)$. Signal degradation and precision loss began to appear after this speed, as shown in Figure 6.20(b) for $30RPM$. One explanation of this is the sensor ranging measurements started to miss the target. In these cases, it was noticed that the rotation rate was faster than the sensor's refresh rate (it was 27ms in short-range continuous mode).

$$Radial\ Coverage\ \zeta = \left(1 - \frac{dropped\ nodes}{total\ nodes}\right)\% \tag{6.2}$$

In Figure 6.21, it is observed that for 8cm targets 1m away, the average valid target acquisition lay within 10cm of the desired 1m distance. This resulted in having less than

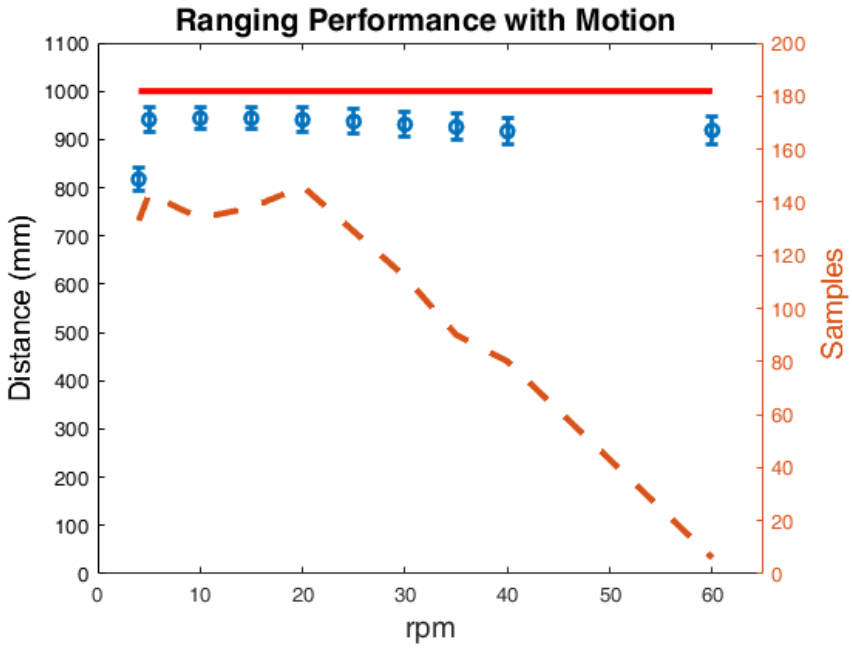


Figure 6.21: Red line representing the actual distance away from the target, Blue markers are the average distance recorded with error, Orange Line representing the samples identifying the target in 15 seconds per RPM.

10% error in the distance for $5RPM$ to $60RPMs$ readings. All data points between $4RPM$ to $20RPM$ contained more than 130 samples. This trend visually identified the end of the stable ranging motion region, which resided at $20RPM(120°/s)$.

To highlight the significance of this value, the fastest angular joint velocity a UR10 robot can rotate is $180°/s$ or $30RPM$. Though there were inconsistencies in the $30RPM$ test, it did not exceed more than three sensor pulse skips (Figure 6.20(b)). This meant that at any given point, there was an 81.25% radial coverage ζ as seen in Eq. (6.2). Compared to the Prototype v1 with 8-sensor nodes 62.5% radial coverage, the Prototype v2 design generated an 18.75% increase in radial workspace coverage at max operating speed near the $1.2m$ sensor ranging limit.

An explanation for a decrease in precision at higher speeds could be in the way the data from the sensors was acquired. In these tests, the system was set to report pseudo-synchronously. This means the system would not report a capture until all sixteen sensors had new, valid data points. This limitation was applied to these tests to maintain the approach of testing the sensor system in the most challenging application environment. If all

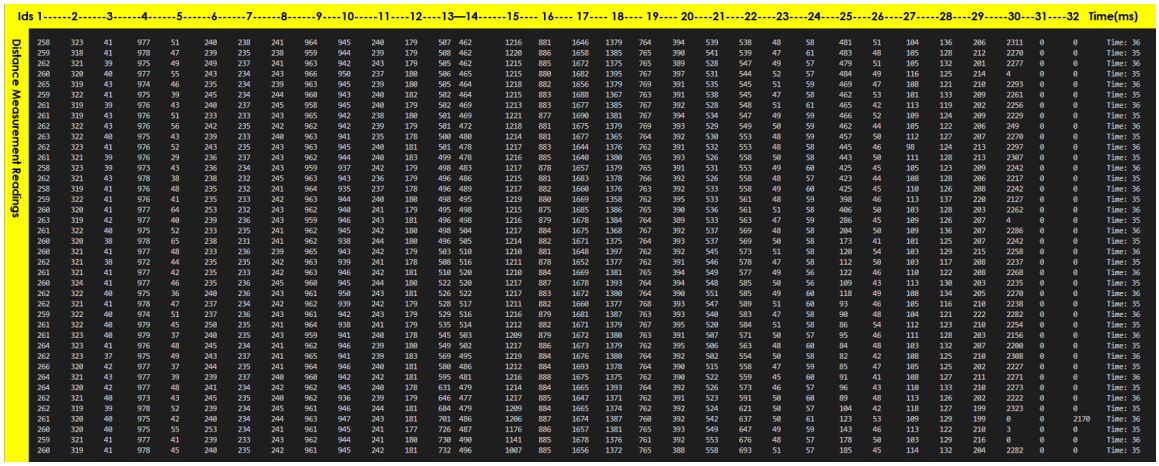


Figure 6.22: The output of the load and data acquisition speed test for the 32-node ToF configuration of Prototype v2.

nodes were set to acquire its data asynchronously, each sensor would be able to operate at max speed, allowing each one to take more samples in the given time. This would allow for the averaging of the sensor to acquire the minimum amount of accurate results.

In the nest section, we present the results of data acquisition test for the 32-node ToF configuration of Prototype v2.

6.2.4.4 Full 32 ToF Sensor Node Setup validation

The data acquisition for a maximum 32-node ToF sensor array was validated, and the output of the of the load and data acquisition speed test is shown in Figure 6.22). The resulting data acquisition speed had an average acquisition frequency of $27.778Hz$ for the pseudo-synchronous operation. This speed is the lower bound of the maximum data acquisition that can be achieved.

$$I2C\ Transmission\ Loss = \left(\frac{transmitted\ bits}{I2C\ data\ rate}\right) \tag{6.3a}$$

$$Serial\ Transmission\ Loss = \left(\frac{transmitted\ bits}{Serial\ data\ rate}\right) \tag{6.3b}$$

$$Total\ Loss = I2C\ Transmission\ Loss + Serial\ Transmission\ Loss \tag{6.3c}$$

$$= \left(\frac{576\ bits}{100000\ bps}\right) + \left(\frac{546\ bits}{115200\ bps}\right)$$

$$Frame\ Period = loss + programmed\ timing\ budget \tag{6.3d}$$

A frame of data is defined as a new distance measurement from all 32 sensor nodes. The frame period calculated was 37.5ms. This period of acquisition is due to the losses from I2C and serial transmission. I2C transmits eight IO expander statuses and 32 sensor readings per frame. When broken down to bits, the I2C transmission loss is 576 bits per frame (transmitted bits) at 100 kbps (I2C data rate) (according to Eq.6.3a). Once all data is collected, these values are sent in a formatted serial transmit of 546 bits at 115200 bps Eq. (according to Eq.6.3b). When these transmissions per rate are converted to time, the total loss is 10.5ms (according to Eq.6.3c). Thus, 10.5ms is lost in acquiring and transmitting the data each frame. The timing budget for the sensors were set to 27ms per initialization sequence. Hence, the total frame period according to Eq. 6.3d is 37.5ms frame period is found Eq. (6.3d).

The 32-node ToF sensor data acquisition also allowed for observation of the setup under full load. The ToF sensor array was estimated to pull near 3A of current. This was based on the max current draw of a single ToF sensor being 20mA. However, during full system acquisition, the sensor array only drew 0.41A of current. The actual draw of the system was significantly less than the estimated draw because the estimated draw was based on the worst-case scenario of all sensors ranging at the same time. However, since the software algorithm was designed to read a new distance measurement from each sensor based one at a time based on the availability of the sensor reading, the likelihood of all 32 sensors ranging at the exact same time is close to zero. This result showed that the actual load requirements for the 32-node ToF sensor array were significantly less than anticipated and meant that the system would require significantly less external power.

6.2.5 Limitations

The measured load range of the ToF sensor array is small enough that the system could be potentially powered from a battery or independent power source instead of requiring power from the robot or other external sources. The PCBs used could be flexible PCBs for more compact and smaller footprint of the ToF sensor array.

Unlike the Prototype v1, the measurements from the sensor were not low pass filtered,

and only distance data was collected. The error codes, accuracy, and signal strength information for each sensor were not transmitted. As the size of the data packet increases, it would add to the losses from I2C and serial transmission.

The design can be further improved by using FPGAs that would provide faster and parallel processing of the sensor data. Other sensor types, such as the upgraded VL53L1X that provides ranging at $100Hz$ could be used. Other ToF sensor chipsets, such as the Terabee [23] and Broadcom AFBR-S50MV85G (16pixels depth at 3kHz over SPI) can be explored . In the future, when mm-wave radars have smaller footprint can be used for distance measurements. These have been used as an extrinsic form of sensing in [48] for safe HRI.

The next step in this research is upgrading the hardware and software in compliance with industrial standards. The IEC 61496-3:2018 [127] specifies additional requirements for 'the design, construction and testing of electro-sensitive protective equipment (ESPE) designed specifically to detect persons or parts of persons as part of a safety-related system, employing active opto-electronic protective devices responsive to diffuse reflection (AOPDDRs)'. This would require the inclusion of redundancy measures, error logging, and monitoring the state of the ToF sensor modules and sensor arrays. Test setups of common HRC scenarios such as coexistence in the workspace while pick and place should be implemented with the Prototype v2. A risk assessment as per ISO 10218 [12] and ISO 13849 safety [13, 128] for machinery standard could be performed.

In the following section, an analysis of the sensing coverage of simulated ToF sensor arrays for varying on-robot placements is presented.

6.3 Sensing Volume Coverage of ToF Sensor Arrays

Each array is considered as an augmentation to the robot body such that each observation incoming from an array is interpreted as an extension of the kinematic chain of the robot. This enables the sensing strategy to leverage the robot motion and provide exclusive coverage from the areas in the workspace where the robot is, and headed to. The setup also allows flexibility in terms of on-robot placement; the arrays can be positioned anywhere on the robot links to achieve optimal sensing coverage. Unlike 2D scanning LiDARs, that provide planer information of separation and relative speed, this approach provides a 3D

information with respect to the robot joint positions.

Determining the minimum distance between two bodies is a non-trivial problem and central to any speed and separation monitoring methodology. During SSM, the ToF sensor arrays placed on the robot should be able to detect a more substantial part of the robot's workspace for accurate distance measurements. The study presented in this section analyzes the volume of the ToF sensor arrays' detection zone and its coverage of the robot's workspace. This volume is termed as the *sensing volume coverage*.

Here, we present a methodology for volumetry using octrees to quantify the detection/sensing volume of the sensors [2, 129]. Based on this methodology, the study analyzes the sensing volume coverage of the robot's workspace as well as the shared human-robot collaborative workspace for various configurations of ToF sensor arrays. The calculated detection/sensing volume coverage can be used to determine the choice of placement of ToF sensor arrays on the robot based on the task-specific human-robot interaction.

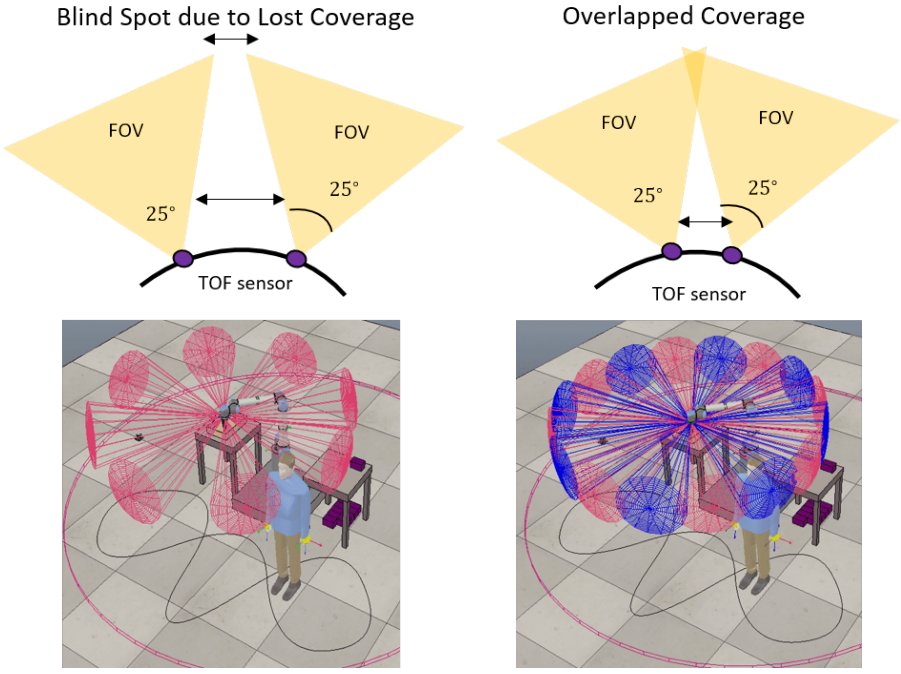
This study leverages the digital-twin setup of the HRC task and the simulated ToF sensor arrays placed on the robot in varying configurations. In the next section, we present the ToF sensor array setup and configurations analyzed in this study.

6.3.1 Time-of-Flight Sensor Array Setup

In this study, we analyze the sparsity of ToF sensors per array and the number and placement of rings on the robot links and its effect on the sensing volume coverage. Each ToF sensor node in the ToF array is a single unit solid-state lidar with a maximum detection range of $1.5m$ and a field-of-view (FOV) of $25°$ degrees. The sensing volume of each sensor in a ToF array can be represented as a cone of height $1.5m$ and angle $25°$, as shown in Fig 6.23. More details about the sensor setup and its use for safer HRC can be found in our previous works [1, 6].

Different ToF sensor configurations are used to quantify the effect of blind-spots on sensing volume coverage. For brevity and ease of reference, different ToF sensor setup configurations, as shown in Figure 6.23 and 6.24 are represented as follows:

$$n\{i\}\_\{j\}\_\{\theta\}$$



(a) (b)

Figure 6.23: (a)The ToF sensor arrays/rings with eight sensor nodes i.e. *n1_8_0*. There is a loss of coverage both far and near the robot. The simulated representation of the ToF array/ring mounted on the last wrist-3 joint(*Tool*) of the UR10 robot is shown (bottom). (b) The ToF sensor arrays/rings with sixteen sensor nodes i.e. *n1_16_0* showing overlapped coverage to compensate the lost coverage.

$$where \begin{cases} i & \text{is num. of rings on the shoulder and elbow links} \\ j & \text{is num. of sensors per ring.} \\ \theta & \text{is the tilt angle of a sensor} \\ & \text{w.r.t to the center of the ring.} \end{cases}$$

In the next section, we define different sensing volumes quantified for this analysis.

6.3.2 Sensing Volumes

Maximum Sensing Volume V_{max} is the ideal workspace in cartesian coordinate that the sensors should cover around the robot to ensure safe human-robot interaction. Sensing volume coverage is the subset of this volume that is covered by the FOV of the ToF sensor

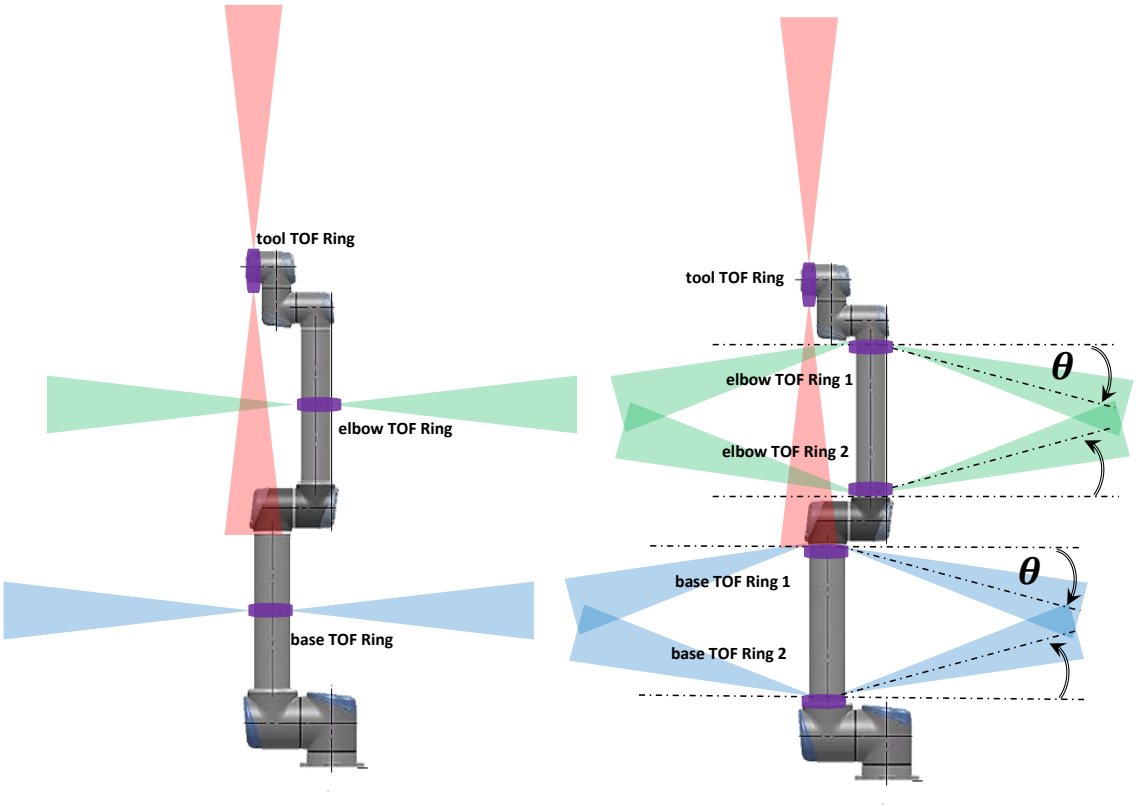


Figure 6.24: The sensor configuration for single i.e. *n1_16_0* and double arrays i.e. *n2_16_θ* (on shoulder and elbow links of UR 10) to measure sensing volume coverage. This configuration aims to cover volume near the robot.

rings. Maximum sensing volume differs based on the task, the application, and the amount of human-robot interaction. In this work, four different maximum volumes are suggested, and are shown in Figure 6.25. They are described as follows:

- **Operating Workspace Volume** (V_O): This is the operating workspace of the robot. The maximum reaching workspace of the UR10 robot used here is a sphere of radius $1.3m$. Sensing Volume Coverage of V_O can be used to determine how much the ToF sensors cover near the robot. For tasks that require close human proximity to the robot, the ToF setup that gives maximum coverage of this volume can be considered.

- **Tool (Tool Control Point -TCP) Volume** (V_T): This is the sphere centered around the TCP of the robot V_T. Here the sphere radius is the maximum detection range of the

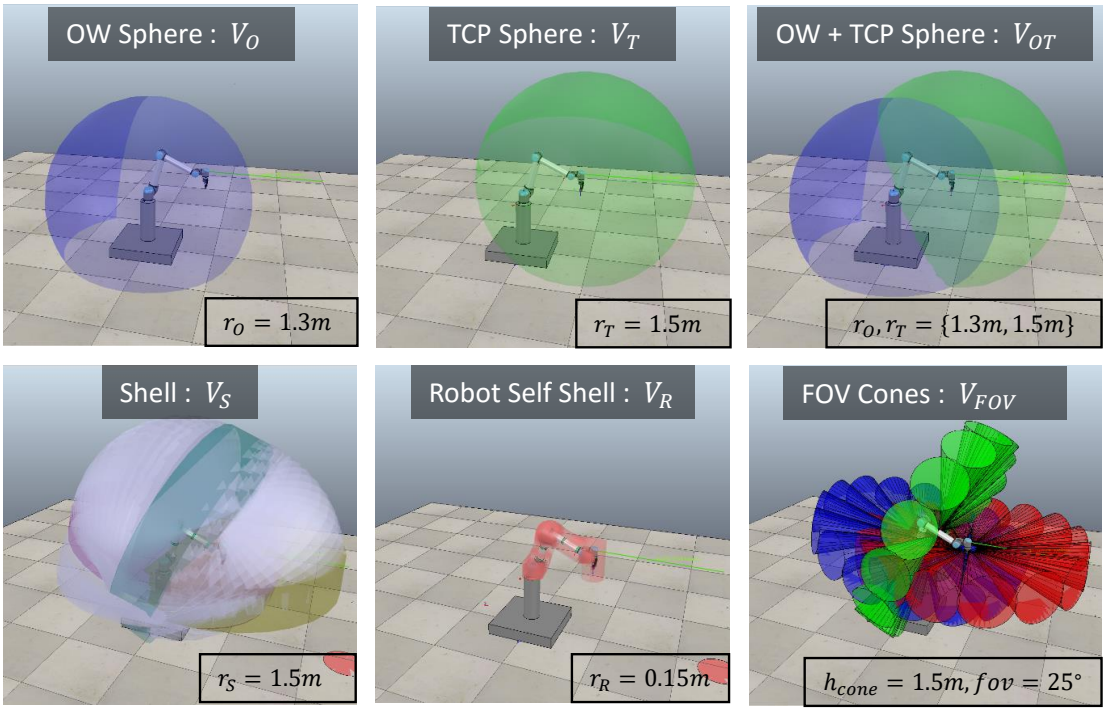


Figure 6.25: Maximum Sensing Volume $V_{max} = \{V_O, V_T, V_{OT}, V_S\}$ used to quantify the sensing volume coverage of sensors with a total sensing volume coverage represented as V_{FOV}. The volume occupied by the robot self V_R, that is subtracted during Sensing Volume Coverage analysis.

ToF sensor i.e., $1.5m$. This workspace changes based on the TCP position. The TCP velocity and distance from TCP to human is mainly considered of safety in HRC [46] [43]. Hence, for scenarios where the robot performs Speed and Separation Monitoring (SSM) [1], coverage in this volume space can be used to choose a ToF sensor configuration.

- **Operating Workspace + Tool Volume** (V_{OT}): This is the combined volume in the workspace. To determine the optimal ToF Sensor configuration that gives coverage for far and near volumes of the robot, the coverage in this combined volume can be used.

- **Shell Volume** (V_S): This is a tubular volume or a shell of a fixed radius. The shell is defined along the curved axis comprised of all robot link endpoints. The curved axis starts at the base link and ends at the TCP. This shell represents a more exact volume for which the sensing volume coverage should be maximized. This workspace changes based on the robot pose.

FOV Volume for a sensor node i on ToF array j can be written as ${}^{ij}V_{fov}$. The combined sensing volume $V_{FOV} = \sum({}^{ij}V_{fov})$ for $\forall(i, j)$ (which is for all ToF sensor nodes in all the ToF sensor arrays). The overlap of this volume with the volumes described above is used to determine the sensing coverage of a ToF sensor setup configuration.

Inner Volume of the Robot (V_R) can be defined by the space occupied by the robot. Here we approximate it as a shell around the robot. In this work, for UR10 a shell of inner radius $0.15m$ is assumed (based on the maximum width of the bounding box of the largest shoulder link of UR10). This volume space is subtracted from all volumes.

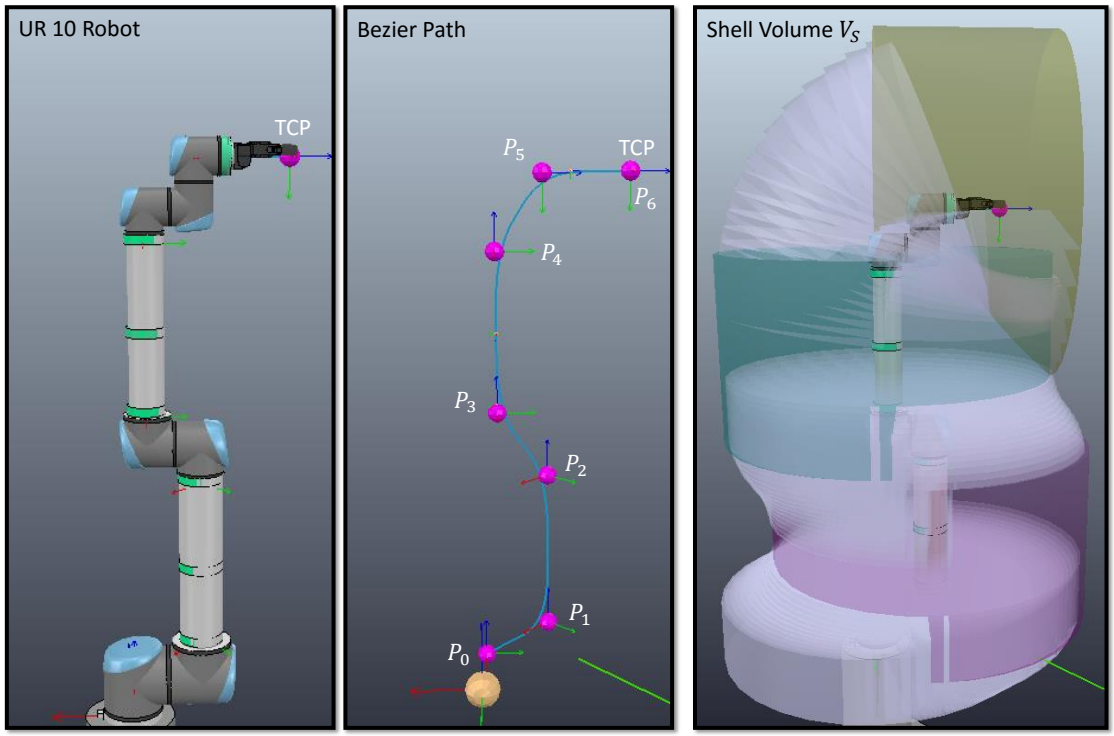


Figure 6.26: Generating Shell Volume (in the image of radius $0.5m$) along the Bezier curve $r(t)$ generated by the UR10 robot link endpoints $P_0, P_1..P_6$. The Bezier interpolation is represented as the gray, and where no interpolation was done is represented with a different color.

In order to calculate the Shell Volume, Bezier interpolation of the robot pose using the endpoints of the robot links is done, refer Figure 6.26. This is detailed further in the following sections.

6.3.2.1 Robot Pose as a Bezier curve

For this work, a piece-wise Bezier curve approach has been used to generate a curve representing the robot pose. It essentially means part of the line segments between two points is not interpolated if it is co-linear, as shown in Figure 6.26. This was helpful as the interpolation was needed around the joints of the robot. A piece-wise Bezier interpolation

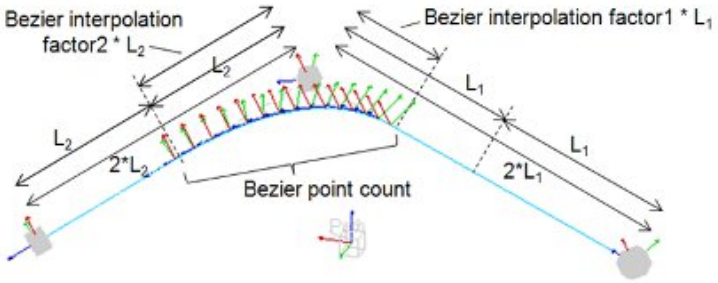


Figure 6.27: The Bezier Curve Interpolation for defining the curve given three control points P_0, P_1, P_2 [29].

determines at what point on the line segment between two control points does Bezier interpolation needs to be done. This is defined based on Bezier interpolation factors and the number of interpolation points (refer Figure 6.27). The readers can refer to V-REP API [29] and [130] for more details.

6.3.2.2 Calculating the Shell Volume

This volume can be calculated by rotating a solid revolution along a curve. This is also known as 'Curved Axis Solids of Revolution' [131] (refer Figure 6.28). This can be formulated as :

$$\begin{aligned} V_S &= \pi \int_t \Big((\boldsymbol{f}(\boldsymbol{t}) - \boldsymbol{r}(\boldsymbol{t}))^2 - (\boldsymbol{g}(\boldsymbol{t}) - \boldsymbol{r}(\boldsymbol{t}))^2 \Big) dt \\ &= \pi \int_t \Big((R_{outer} \bullet \hat{\boldsymbol{n}}(\boldsymbol{t}))^2 - (R_{inner} \bullet \hat{\boldsymbol{n}}(\boldsymbol{t}))^2 \Big) dt \end{aligned} \tag{6.4}$$

Alternatively according to Pappus Centroid Theorem [132] as

$$V_S = \pi (R_{outer}^2 - R_{inner}^2) \bullet arclength(\boldsymbol{r}(\boldsymbol{t})) \tag{6.5}$$

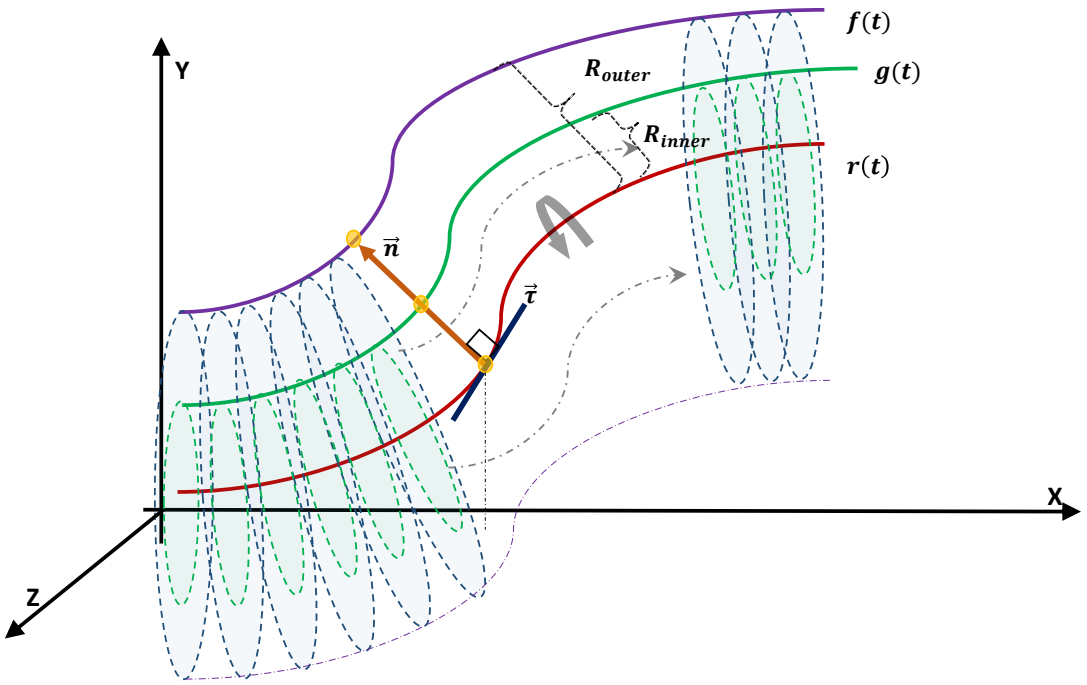


Figure 6.28: The shell volume calculation using the washer-method for curves $f(t)$ and $g(t)$ around $r(t)$ - Curved Axis Solids of Revolution.

where

$$\text{Length of Curve : } arclength(\boldsymbol{r(t)}) = \int_t \|\boldsymbol{r(t)}'\| dt \tag{6.6a}$$

$$\text{Outer Shell Vector : } \boldsymbol{f(t)} = \boldsymbol{r(t)} + R_{outer} \bullet \hat{\boldsymbol{n}}(\boldsymbol{t}) \tag{6.6b}$$

$$\text{Inner Shell Vector : } \boldsymbol{g(t)} = \boldsymbol{r(t)} + R_{inner} \bullet \hat{\boldsymbol{n}}(\boldsymbol{t}) \tag{6.6c}$$

$$\text{Normal Unit Vector : } \hat{\boldsymbol{n}}(\boldsymbol{t}) = \frac{\boldsymbol{\tau(t)}'}{\|\boldsymbol{\tau(t)}'\|} \tag{6.7a}$$

$$\text{Tangent Unit Vector : } \hat{\boldsymbol{\tau}}(\boldsymbol{t}) = \frac{\boldsymbol{r(t)}'}{\|\boldsymbol{r(t)}'\|} \tag{6.7b}$$

Determining the volume covered by the shell above can be computationally expensive and difficult to quantify, especially with intersections of other volumes. Hence, an approximation using octree-based volumetry has been made [129].

6.3.3 Octree-based Volumetry

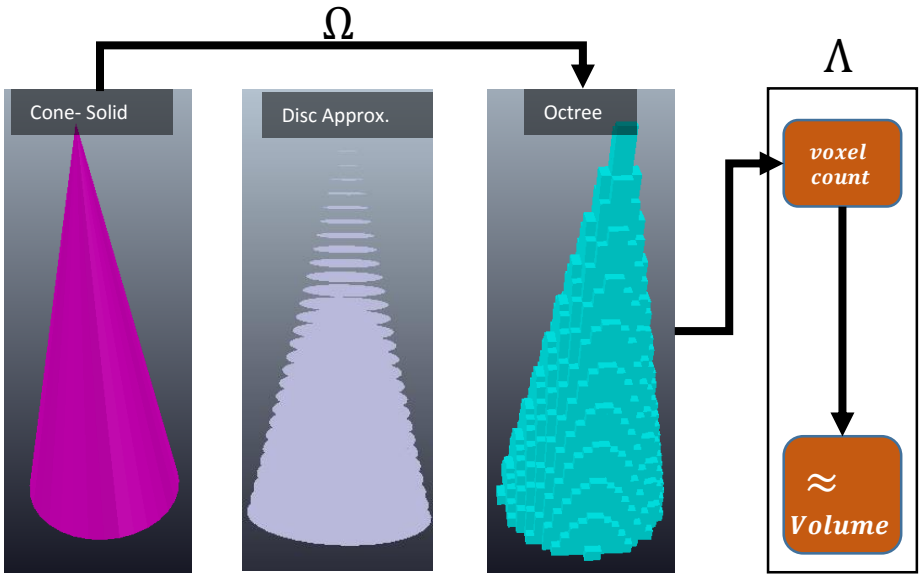


Figure 6.29: Octree-based volumetry pipeline for a Cone shape. The Ω function represents a given shape as an octree and the Λ operator quantifies the volume occupied by the octree. Volume of Cone is $0.173m^3$ and volume reported by Octree is $0.17m^3$.

Octrees are used for the voxel representation of any shape in a 3D cartesian space. Octree representation of volume region V_{max} is represented as $\Omega(V_{max})$. The volume can be calculated from octree by counting the number of voxels in $\Omega(V_{max})$. A voxel is a cube with side length l_{voxel} then the volume of the region occupied by octree $\Omega(V_{max})$ can be written as $\Lambda(\Omega(V_{max})) = voxelcount \cdot l_{voxel}^3$. Octrees can be used to merge or subtract voxels from other octrees. As the V-REP represents shapes as hollow, the inner volume of the shape needs to be taken into account. Hence, before a shape is converted into octrees, it is decimated into discs of varying radius spaced by voxel size l_{voxel}. An octree-based volumetry pipeline for a FOV cone of a ToF sensor is shown in Figure 6.29.

Next, the formulation for calculating the sensing coverage for a given ToF sensor configuration is shown.

6.3.3.1 Coverage of a ToF sensor configuration

Given a maximum coverage volume $V_{max} = \{V_O, V_T, V_{OT}, V_S\}$, the coverage ζ, of a ToF sensor configuration ($n\_i\_j\_\theta°$) with the field-of-view volume of V_{FOV} can be written as:

$$\zeta(\%) = \frac{\Lambda\left(\Omega(V_{max}) \cap \Omega(V_{FOV})\right)}{\Lambda(\Omega(V_{max}))} \times 100 \tag{6.8}$$

Alternatively, as V-REP allows only set addition and subtraction of voxels from Octrees, Equation 6.8 can be re-written as:

$$\zeta(\%) = \frac{\Lambda\left(\Omega(V_{max})\right) - \Lambda\left(\Omega(V_{max}) - \Omega(V_{FOV})\right)}{\Lambda(\Omega(V_{max}))} \times 100 \tag{6.9}$$

A Venn diagram representation of the octree-based calculation of sensing volume coverage is shown in Figure 6.30.

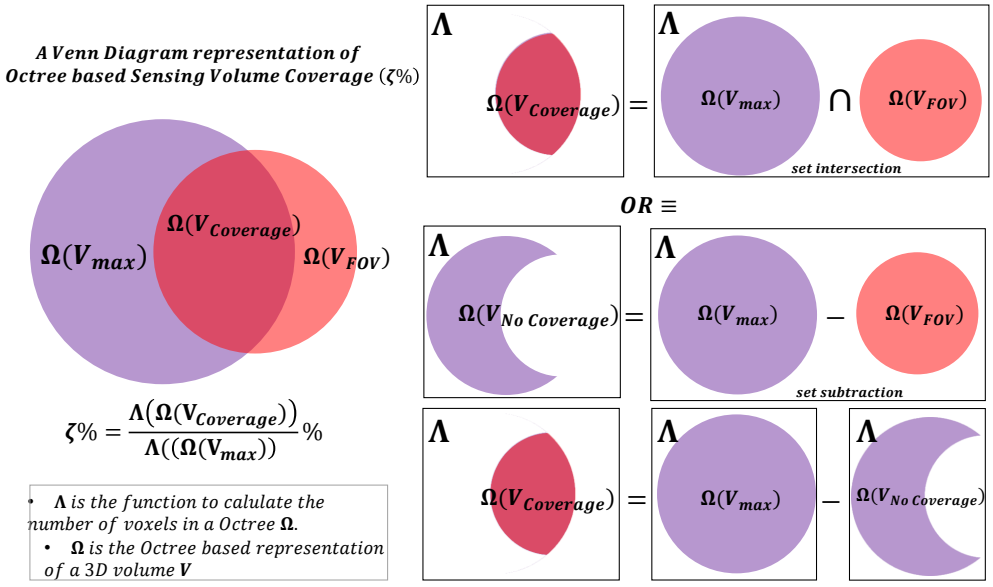


Figure 6.30: A Venn Diagram representation of the octree-based sensing volume coverage ($\zeta\%$) for the maximum robot workspace volume V_{max} and the ToF sensor arrays sensing volume V_{FOV} for a given configuration.

6.3.4 Setup for measuring the Sensing Volume Coverage

The experiment setup is a generic robot pick and place task of placing ten products in a box, as described previously in Section 4.1.3.

In this work, the coverage at the robot pose when the robot is least safe i.e. moving at the highest speed during the task is measured. This was done with the reasoning that the ToF sensor arrays have the maximum coverage to detect and anticipate human/operator in the workspace. This setup is task-specific but can be extended to any task which requires coverage either near or farther from the robot based on the human-robot interaction during the task. Hence, in this study, different V_{max} volumes are considered that represent ideal maximum coverage both near and farther from the robot. For this study, the octree-based volume calculations were done using V-REP [29].

In Speed and Separation Monitoring (SSM) based collaborative tasks [46], the minimum distance calculations and directed velocities of the human and the robot are generally done with respect to the base and TCP of the robot. That is why the sensing volume coverage in a sphere representing the operating workspace (V_O) and a detection sphere centered at the TCP (V_T) are analyzed. However, according to the ISO standards [43] and also in other works [60], the minimum distance and directed speeds can be with respect to any point on the robot. So a more exact sensing volume coverage is analyzed using a shell (V_S) around the robot self, that changes with the robot pose.

The sensing volume coverage is measured by determining the overlap of the ToF sensor arrays volume, V_{FOV} with the maximum ideal volume V_{max}, which can be $V_{max} = \{V_O, V_T, V_{OT}, V_S\}$ (refer Section 6.3.2).

6.3.4.1 Sensing Volume for ToF Sensor Arrays

The sensor detection volume of a ToF laser ranging sensor is modeled as a cone with a field-of-view given by the beam angle of 25° degrees and detection range i.e. the cone height as $1.5m$. To verify that the detection volume can be approximated as a cone, a simple test of projecting the laser beam emitted by a ToF sensor on a whiteboard was done, the image of the projection enhanced, and the contour of the projection was checked. As shown in Figure 6.31 it can be seen that the projection shape approximates to a circle, which validates the modeling of ToF sensor detection volume as a cone. It can be seen in Figure 6.32 the

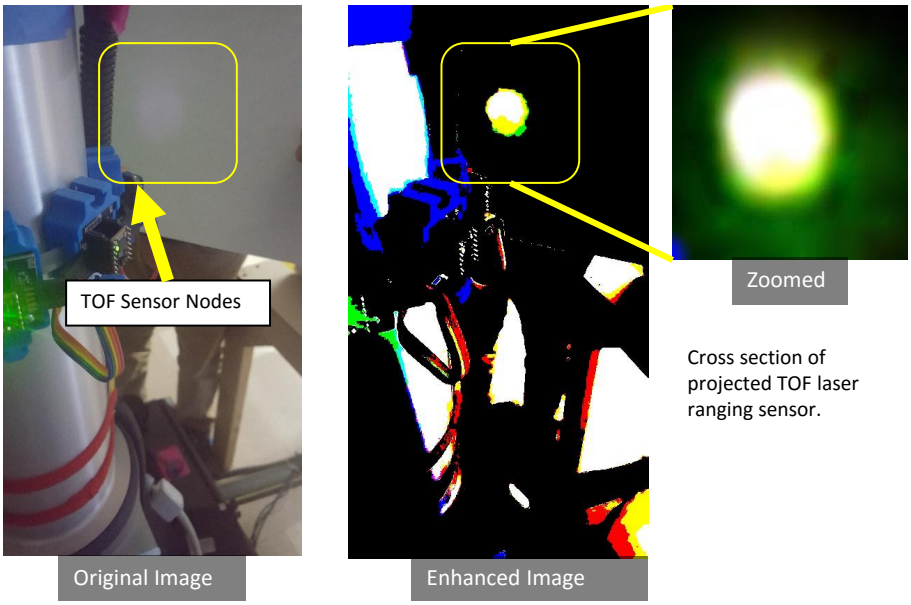


Figure 6.31: Verification of the Time-of-Flight sensor node sensing volume can be modeled as a cone.

different V_{FOV} sensing volumes for different ToF sensor configurations.

6.3.4.2 Sensing Coverage Measurements

In order to analyze and compare the sensing volume coverage, the following measurements were taken:

- Impact on Sensing Coverage for ToF Configurations with a different number of rings for all V_{max}, as shown in Figure 6.32-Top Row. The configurations compared were for single rings on elbow and shoulder robot links with eight and sixteen sensors per array ($n1\_8\_0°$, $n1\_16\_0°$), dual rings with varying $\theta° \in \{10°, 25°, 55°\}$ ($n1\_16\_\theta°$)and also three rings with ($n3\_16\_55°$) which is mounting rings at the end of robot links at an angle 55°and also the center of the robot link.
- Sensing Coverage for ToF Configurations $n2\_16\_\theta°$ with varying θ for all V_{max}. The $\theta°$ is varied 5° in the range of $\theta° \in [0°, 60°]$. This is to measure the impact of change in $\theta°$ to the coverage in the near and farther zones of the robot.
- Sensing Coverage for ToF Configurations $n2\_16\_\theta°$ with varying θ for shell volume V_S with varying radius $r_S \in \{1.5m, 1.1m, 0.9m, 0.7m, 0.5m\}$ (examples shown in Figure 6.33).

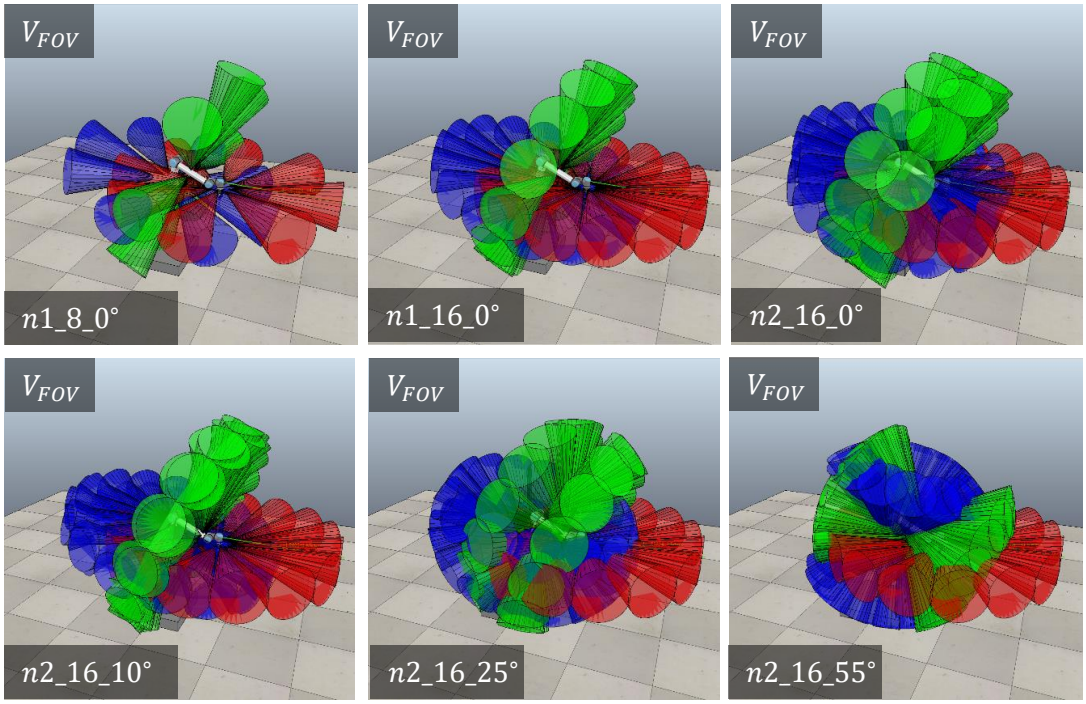


Figure 6.32: The ToF Sensor rings setups (Top Row) shows the 3 major ToF sensor configurations for $n1\_8\_0°$ and $n1\_16\_0°$ as single rings with eight & sixteen sensors, respectively, and $n\_2\_16\_0°$ - dual rings on shoulder and elbow links of the robot. The ToF Sensor rings setups (Bottom Row) shows the angle variation of sensors on the array for $n2\_16\_\theta°$ where $\theta° \in \{0°, 25°, 55°\}$.

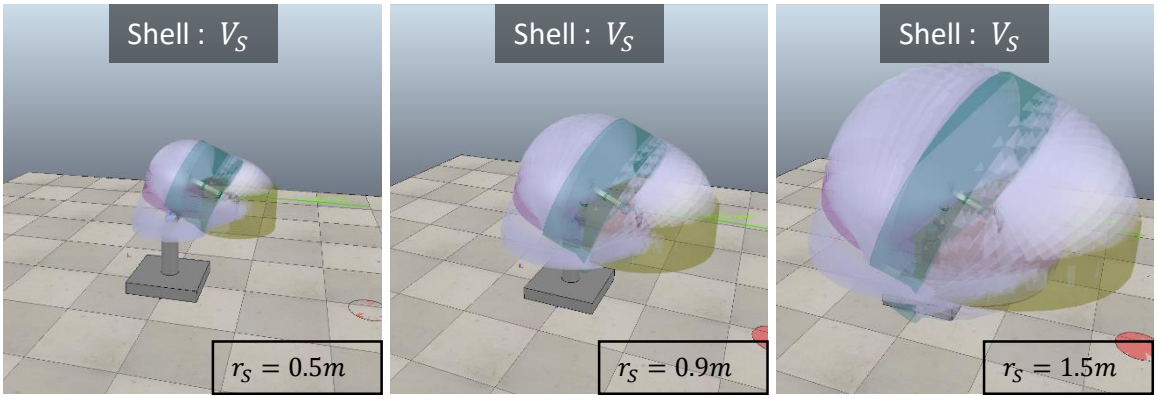


Figure 6.33: The ToF Sensor rings setups $n\_2\_16\_\theta°$ are used to determine coverage for changing V_S with varying radius r_S. The figure shown are $r_S \in \{0.5m, 0.9m, 1.5m\}$.

The results of these comparisons are shown and discussed in the following section.

6.3.5 Results

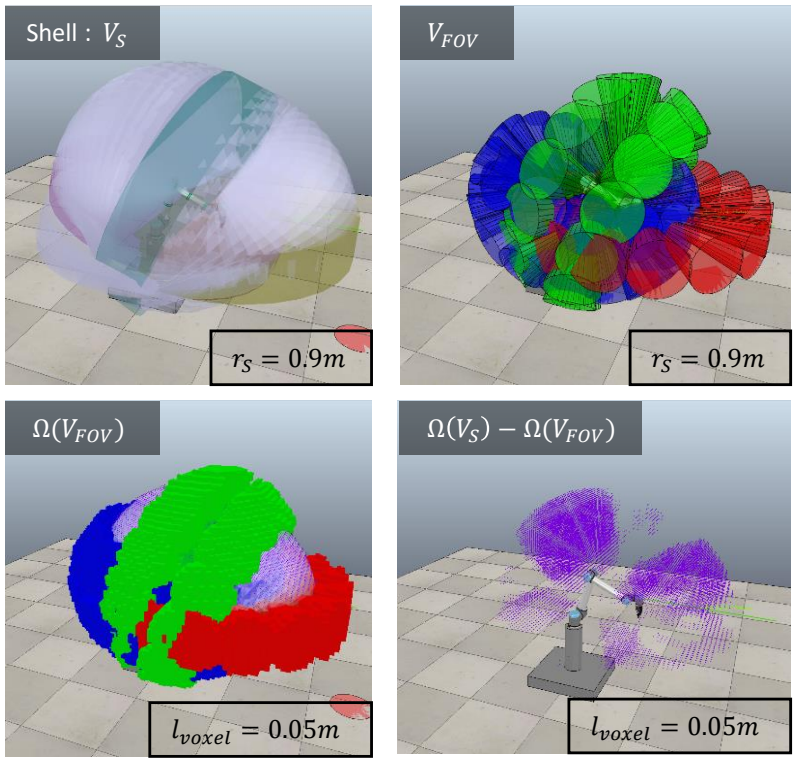


Figure 6.34: An example of octree-based approximation for calculating sensing volume coverage for shell volume $V_{max} = V_S$ of radius $r_S = 0.9m$ (Top Left). The V_{FOV} of configuration $n2\_16\_25°$ (Top Right) octree approximation (Bottom Left). The volume not covered by the sensors in the shell (Bottom Right).

Octree-based approximations were used to calculate the sensing volumes $V_{max} \in \{V_O, V_T, V_{OT}, V_S\}$ and also the ToF Sensor array volume V_{FOV}. A measurement for a shell $V_{max} = V_S$ of radius $r_S = 0.9m$ is shown as an example in Figure 6.34. In Figure 6.34(Top Left) V_S is shown, where the gray discs represent the Bezier interpolated points, whereas the straight links are represented with other colors (refer Section 6.3.2.1). The V_{FOV} for configuration $n2\_16\_25°$ is shown in Figure 6.34(Top Right), where red, green and blue cones represent the sensing volume of ToF sensor arrays mounted on tool, elbow and shoulder links of UR10 robot, respectively. The Octree approximation $\Omega(V_S)$ and $\Omega(V_{FOV})$ are shown in Figure6.34 (Bottom Left). For visual clarity, $\Omega(V_S)$ is shown as a violet pointcloud where the points represent the center of the voxels in the octree. The left-over volume of V_S not

covered by the ToF sensor arrays is shown in Figure6.34(Bottom Right). Using Eq. 6.9 sensing volume coverage $\zeta(\%)$ was calculated (see Section 6.3.3).

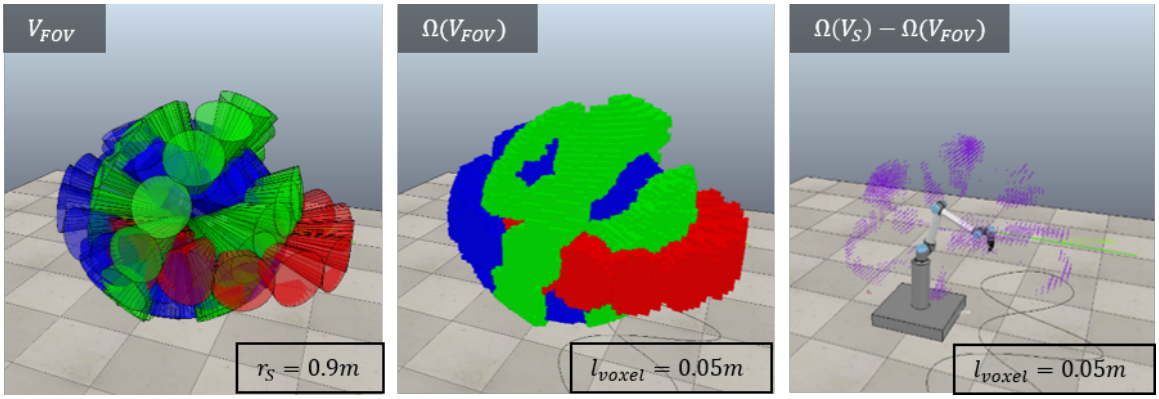


Figure 6.35: An Octree-based approximation for calculating sensing volume coverage for shell volume $V_{max} = V_S$ of radius $r_S = 0.9m$ for configuration $n3\_16\_55°$.

The first set of measurements were done for calculating $\zeta(\%)$ of ToF sensor configurations to observe the impact of increasing the number of sensors per array i.e. $n1\_8\_0°$ to $n1\_16\_0°$, and increasing the number of rings per link i.e. $n1\_16\_0°$, $n2\_16\_\theta°$ and $n3\_16\_\theta°$. The results are shown as a bar-graph in Figure 6.36. The observations were as expected, an increasing coverage with more sensors per array and more rings per link. Another observation that was made is that for $n2\_16\_10°$, the coverage is similar to $n1\_16\_0°$. This is because the coverage of the two rings at 10° overlap (as shown in Figure 6.32), and behave similar to a single array in the center. Thus, change in θ impacts the sensing volume coverage.

To further observe the impact of change in θ in sensing volume coverage $\zeta(\%)$, θ is varied from 0° to 60° for the $n2\_16\_\theta°$ ToF configuration. The results are shown in Figure 6.37. It is observed that as the overlap of the volume V_{FOV} for a given set of ToF sensor arrays on a link increases, $\zeta(\%)$ drops. As observed before, the coverage of $n2\_16\_10°$ is minimum and equivalent to $n1\_16\_0°$. It is observed that the most optimized and maximum coverage is given at $n2\_16\_0°$ and $n2\_16\_25°$ ToF sensor configuration.

In order to observe coverage in the range of $0.5m$ to $1.5m$ from the robot for varying θ in $n2\_16\_\theta°$ ToF configuration, shell-based volume V_S of radius $r_S \in \{0.5m, 0.7m, 0.9m, 1.1m, 1.5m\}$ was considered. In the previous work [1], in the SSM implementation for safety using ToF sensors, $0.5m$ and $1.1m$ were considered as distance thresholds for varying the speeds of

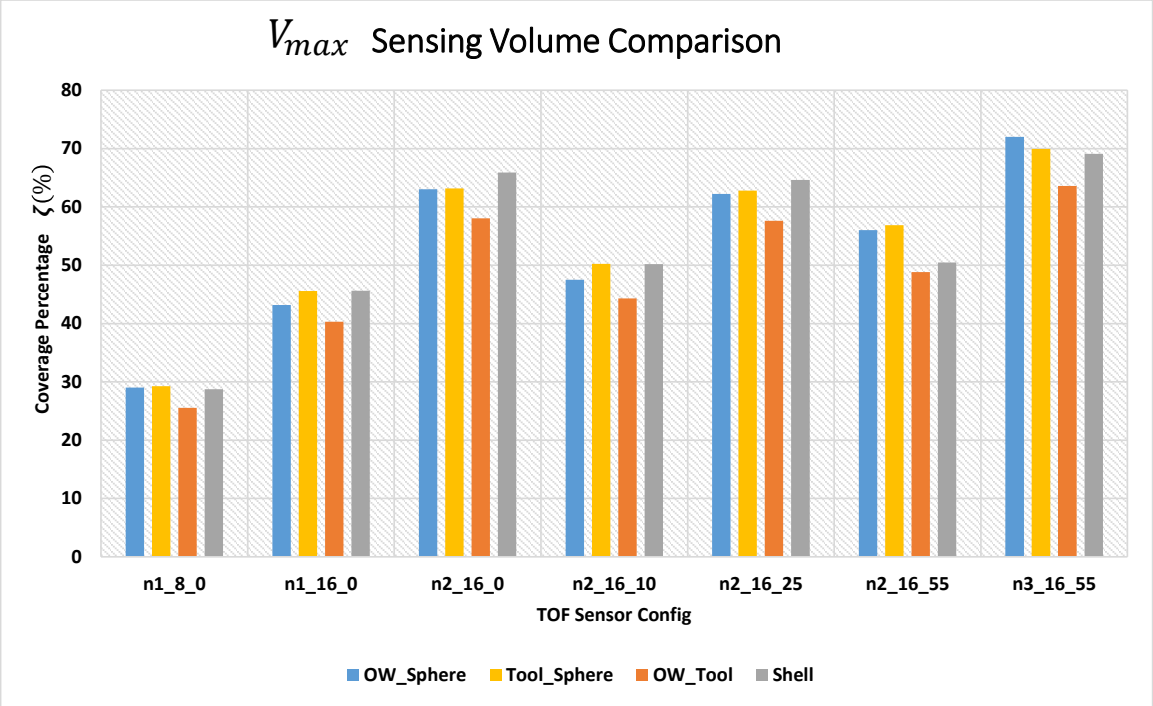


Figure 6.36: Sensing Volume Coverage $\zeta(\%)$ of ToF sensor configurations for all V_{max} to observe impact of increasing number of sensors per array i.e. $n1\_8\_0°$ to $n1\_16\_0°$ and increasing the number of rings per link i.e. $n1\_16\_0°$, $n2\_16\_\theta°$ and $n3\_16\_\theta°$.

the robot. Thereby, if the human is working in close proximity, a sensor configuration that has more coverage closer to the robot can be used. Contrarily for farther distances and better anticipation of human encroaching on the robot workspace, farther coverage becomes important. Hence, sensing volume coverage $\zeta(\%)$ is calculated with varying θ and r_S. The results are shown in Figure 6.38. It can be observed that as θ increases the coverage near the robot also increases. It can be seen that for $r_S = 0.5$ and $n2\_16\_55°$, the coverage $\zeta > 90\%$.

In order to maximize the closer and father coverage, a sensor configuration that combines $n1\_16\_0°$ and $n2\_16\_55°$ i.e. $n3\_16\_55°$, which is placing three rings on the elbow and shoulder links of the robot is also implemented and the sensing volume coverage $\zeta(\%)$ measured. It results in a over 65% coverage for all V_{max} (shown in Figure 6.36), and a coverage of 96.73% and 69.80% for V_S with shell radius r_S of $0.5m$ and $1.5m$, respectively. This is shown in Figure 6.35. The leftover V_S can be compared to Figure 6.34(Bottom Right) to see the difference in coverage.

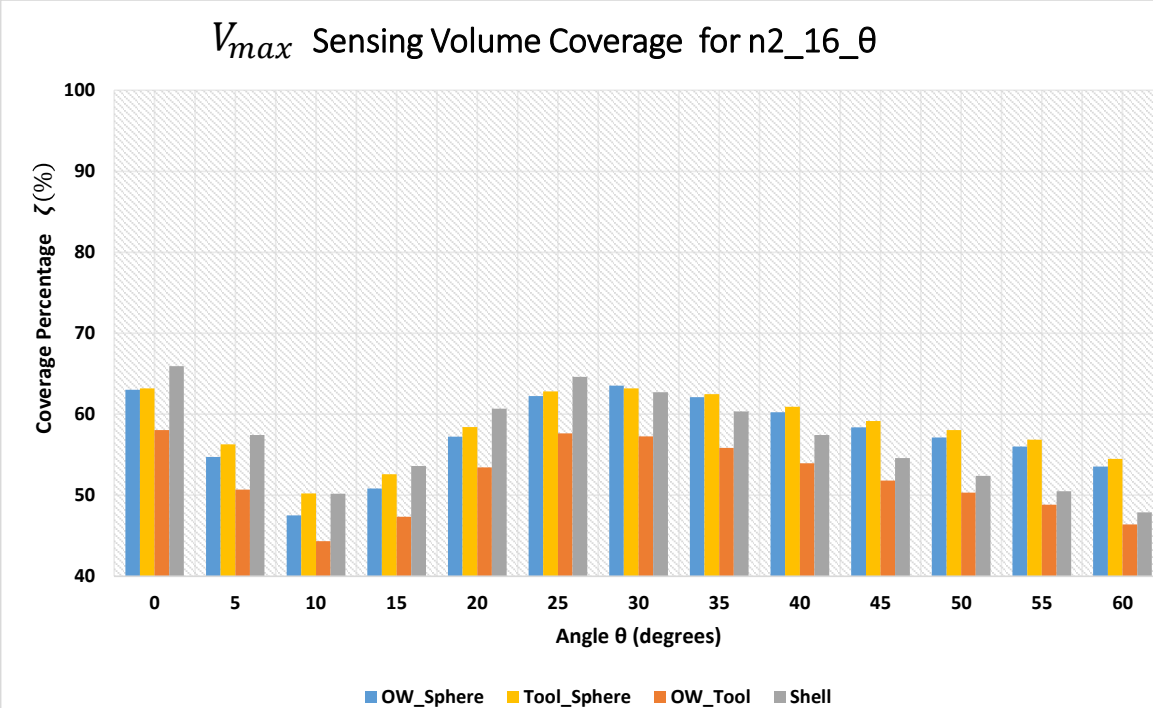


Figure 6.37: Sensing Volume Coverage $\zeta(\%)$ for all V_{max} to observe the impact of increasing θ in ToF sensor configurations $n2\_16\_\theta°$.

The minimum distance accuracy for a human moving in robot workspace for the experiment described in [6] for the ToF sensor configurations mentioned above, is shown in Figure 6.39. Root Mean Square Error (RMSE) and the maximum distance error of the measured minimum distance from the sensors between human-robot, with respect to the ground-truth (the absolute minimum distance, the distance between the closest points on robot and the human) is shown. It was observed that as the sensing volume coverage increases with the number of sensors per array and the number of rings per link, the error decreases.

A video representing this setup can be found at https://youtu.be/SKKwBdk5wlw.

For the same experimental setup as discussed in Section 4.1.3, the task completion times for different ToF sensor placement configurations for a Safety Index algorithm result are consistently around $140sec$, as shown in Figure 6.40. Thus, for an object of the human size there wasn't much difference in terms of productivity.

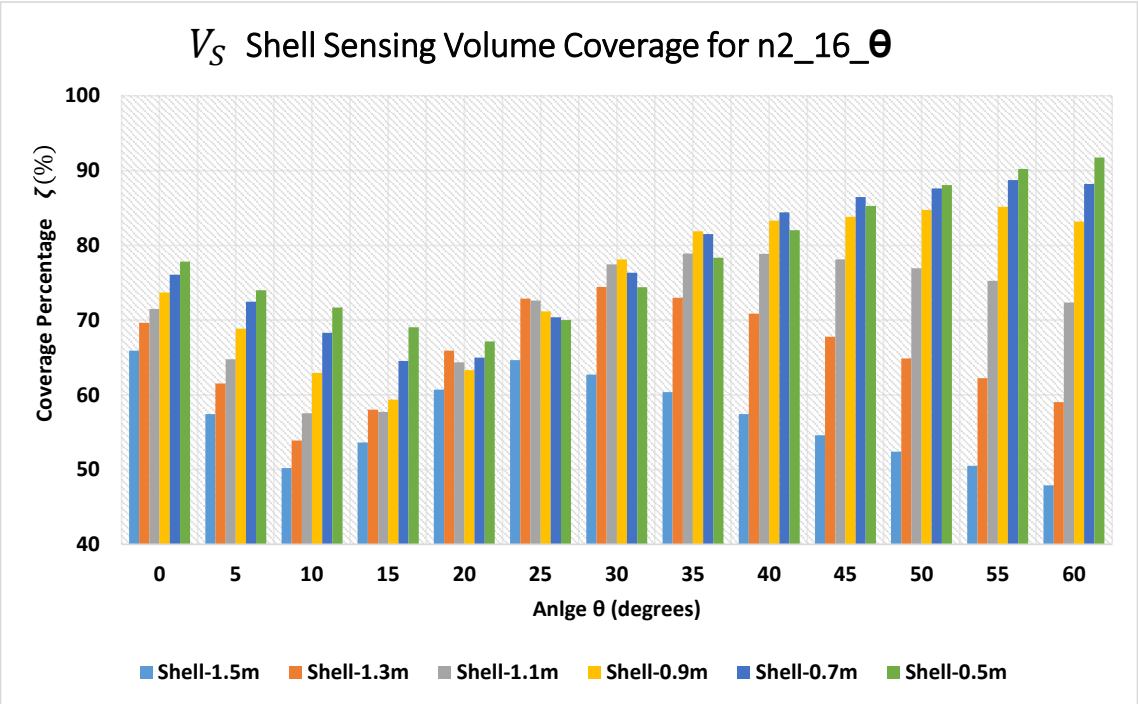


Figure 6.38: Sensing Volume Coverage $\zeta(\%)$ for shell volume V_S with varying radius r_S and varying θ in ToF sensor configurations $n2\_16\_\theta°$.

6.3.6 Limitations

In this section, a methodology to quantify the perception in the form of sensing volume coverage for on-robot ranging sensors is presented. This was shown for various ToF sensor placement configurations, from a varying number of sensor array rings per robot link, and a varying number of sensors per ring. This sensing was done by emulating the sensing properties of the ToF sensor in simulation. It was concluded that increasing the number of sensors per ToF sensor array does help remove blind-spots. However, it was determined that for a human-sized object, the productivity was the same. Thus, there is a need to investigate the performance of safety configurations for different sized objects.

In the next section, we summarize the information presented in this chapter and conclude.

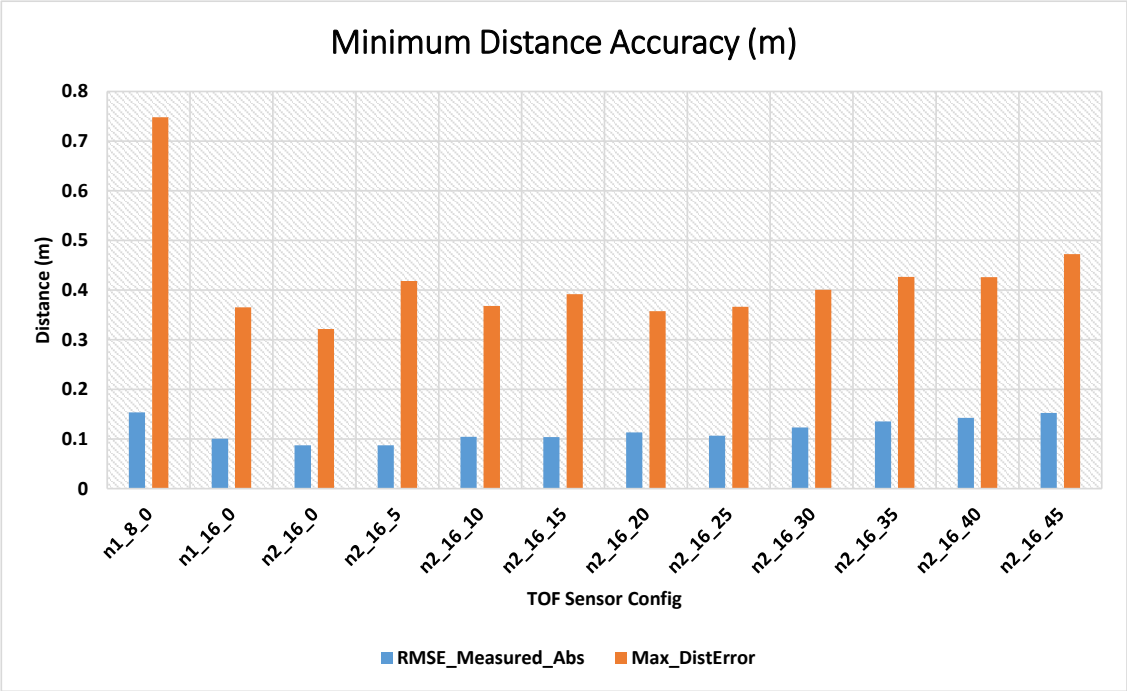


Figure 6.39: Root Mean Square Error (RMSE) and the maximum distance error of the measured minimum distance from the sensors between human-robot, with respect to the ground-truth (the absolute minimum distance, the distance between the closest points on robot and the human), for different ToF sensor configurations.

6.4 Intermediate Conclusion

An 8-node ToF sensor array is implemented to be mounted on the robot center links of the robot. The main drawback is that using only eight nodes results in to blind spots resulting in spikes in the distance measurement. A second prototype with modular design that can be extended up-to 32 ToF sensor nodes is implemented. A sensor characterization analysis in terms of accuracy, overlap coverage and ranging under motion is performed. The sensor showed a consistent output for the worst-case measurement of all new data available at 30Hz. The sensing volume coverage for an 8-node and 16-node sensor array is quantified using octree-based volumetry. It can be concluded that using a 16-node ToF array would help decrease the minimum distance error and solve the blind spot problem. However, for a human detection, the performance for an 8-node and 16-node in a simulation didn't show significant change in terms of productivity.

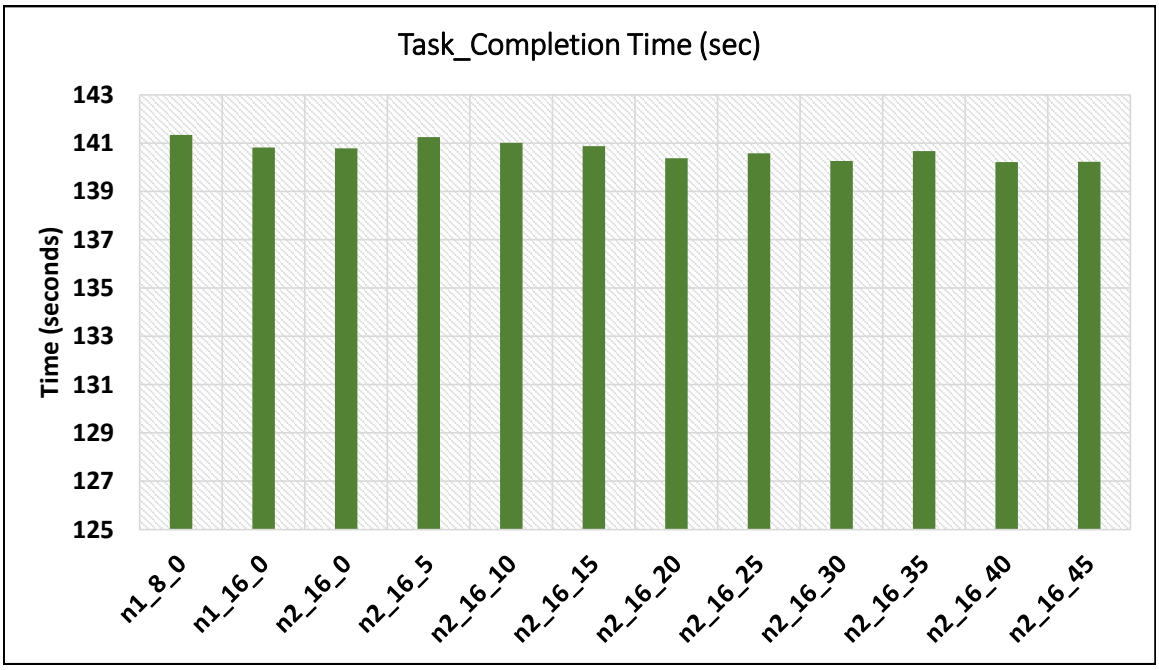


Figure 6.40: Task Completion Time during the experiments for different ToF sensor placement configurations.

In the next chapter, we draw the overall conclusion of the research presented in this dissertation and discuss its impact and future direction.

Chapter 7

Conclusion and Future Work

In this chapter, we present the conclusions and summary of this research, discuss the impact of this research on other studies, and present the future direction of this research.

7.1 Conclusion

The research presented in this dissertation showed the implementation of a dynamic Speed and Separation Monitoring safety approach using Time-of-Flight (ToF) laser-ranging sensor arrays placed at the centers of the links of a robot arm manipulator. The method shown in this research is the first novel safety application for the use of on-robot ToF-based sensor arrays to achieve dynamic Tri-Modal SSM for a robot arm manipulator. This work presented an in-depth analysis of the viability of using on-robot exteroceptive ranging sensors (or as we termed it, intrinsic sensors) for implementing dynamic SSM as a safety measure.

In compliance with the industrial standards, this research showed a dynamic SSM safety formulation for calculation of protective/critical (PSD) and reduced speed (RSD) safety distances. The formulation shown was dependent on the measured human-robot minimum separation distance and the human and robot relative velocities. This research derived a direct computation method with low complexity for measuring the human-robot minimum separation distance, and estimating the relative human-robot speeds using the ToF sensor array measurements and the robot kinematics. The minimum distance calculation based on the ellipsoidal approximation of robot links was validated. It was concluded that the human-robot minimum distance measured from the center of the ToF sensor array rings placed at the center of the robot links was a good approximation to the absolute/ground truth minimum distance.

A motion-capture system with a digital-twin setup was used for validating and analyzing the performance of the ToF-based SSM setup. The ToF-based dynamic SSM was compared to the conventionally used 2D LiDAR-based static SSM and also with the motion-capture and digital-twin-based setup. There were three variations of the SSM safety approaches, TriSSM-Vo, TriSSM-Vr, and Tri-SM implemented. The human-robot distance and speed measurements were done using the 8-node ToF sensor array Prototype v1, the simulated ToF sensors in digital-twin with motion capture, the ideal on-robot sensor in the digital-twin with motion capture, and the simulated 2D LiDAR.

A Cyber-Physical System (a system-of-systems) capable of controlling and monitoring the robot motion, tracking the human movements, and the progress of the human-robot collaboration task was developed. This system was used to analyze the HRC experiments. This system measured the 'actual safety' and the 'perceived safety' of the human-operator working with a robot under varying SSM safety configurations.

In the next section, we highlight the conclusions of our experiments measuring the 'actual safety' of the human-operator in a controlled SSM setup where the human-operator interaction robot was fixed.

On Safety and Productivity of on-robot ToF sensing system

In this research, experiments comparing the 'actual safety' of the human-operator using quantifiable evaluation criteria for measuring the safety, performance, and productivity during an HRC task for an SSM safety configuration were performed. It was concluded that ToF-based dynamic SSM had a significant advantage in comparison to the 2D LiDAR-based SSM. The TriSSM-Vo and TriSSM-Vr safety configurations were over 40% more productive than 2D LiDAR-based Tri-SM setups currently used in the industry.

It was also observed that the TriSSM-Vo based dynamic SSM that considered human-robot relative velocities for estimating the safety was the optimal safety algorithm to be used with ToF sensor arrays. It can be concluded that the on-robot ToF-based dynamic SSM approach can be used as a viable alternative or an addition to the 2D LiDAR-based safety setups in industry. The TriSSM-Vo safety configuration for the HRC tasks was the preferred approach for ensuring human-safety while optimizing productivity. The above conclusions and observations were made for a controlled HRC experiment with fixed human interaction

with the robot, as the human subject walked a predefined path in the shared workspace.

In the next section, we present the conclusions drawn from the human-subject evaluation of the dynamic SSM safety configurations for an HRC setup where the human agent performs an assembly task with the robot.

On Human Subject Evaluation of the Dynamic SSM setup

The results of the experiments suggested that, on average, the human-subject preferred the work with the robot with a TriSSM-Vo safety configuration. The robot behavior with TriSSM-Vr was perceived as the least safe by the human-subjects. The productivity of the HRC task for the Tri-SM setup with ToF Sensor Arrays and Motion Capture sensor modalities was more productive than all 2D LiDAR-based SSM setup. On average, the human-subjects felt safe working with the robot with ToF sensor array-based Tri-SSM safety configuration. It was shown that the humans felt as safe and comfortable working with the robot with TriSSM safety configurations using ToF sensor arrays, as when using the motion capture and digital-twin based measurements. This observation showed that the uncertainty and error in distance measurements did not significantly affect the perceived safety of the human-subject during the task. However, objectively the 'actual' safety was affected by the uncertainty and accuracy of ToF distance measurement.

Hence, to analyze and address the loss of coverage and distance accuracy with the 8-node ToF sensor array prototype, a second prototype with a 32-node ToF sensor array was implemented. Moreover, a study in terms of placement of the ToF sensor arrays to quantify the sensor volume coverage of the robot workspace was performed. An octree-based volumetry methodology for calculating the sensing volume coverage of sensors placed on the robot was proposed and implemented.

To summarize, the main contributions of the research presented in this dissertation are:

- A direct computation method with low complexity for measuring the human-robot minimum separation distance, and estimating the relative human-robot speeds using on-robot ranging sensor arrays and the robot kinematics, for implementation of Dynamic SSM.
- A Cyber-Physical System (a system-of-systems) capable of controlling and monitoring the robot motion, tracking the human movements, and the progress of the human-robot

collaboration (HRC) task during HRC experiments. This system leveraged the digital-twin implementation of the HRC task for analyzing the HRC experiments.

- An octree-based volumetry method for quantifying the on-robot sensor's sensing volume (the volume of the sensor's detection zone) and its coverage of the robot's workspace.
- A modular design and implementation of a 32-node ToF sensor array as a minimum viable product for use as an electronic safety device to implement dynamic SSM with collaborative and industrial robots.

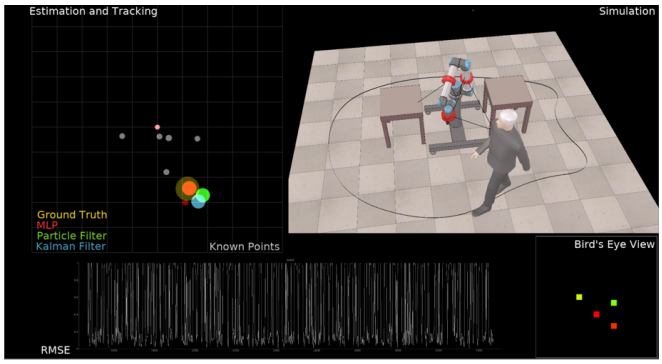
Next, we highlight the impact of this research on other studies.

7.2 Research Impact on Other Studies

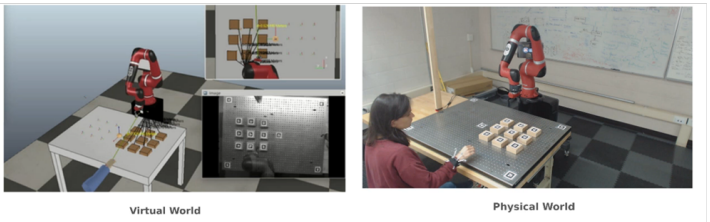
The systems developed and the data generated during the course of this research, have impacted three researches.

- The research in [8, 30] models the uncertainty of the ToF sensor arrays and uses the sparse point cloud data to track the human in the robot workspace. The detection is performed by leveraging an artificial neural network in combination with a Kalman and a Particle filter. It takes a highly sparse 3D point cloud input from the ToF sensor arrays to produce an estimate of the partial pose, which is the ground projection frame of the human footprint. The setup for this research is shown in Figure 7.1(a).
- The research in [9, 31] studies the human-robot interaction for a shared human-robot task. The digital-twin setup inspired by this research is used to observe the change in human behavior under varying cognitive load during the shared HRC tasks. This setup is shown in Figure 7.1(b).
- The research in [3] presents the development of a physiological computing system framework for monitoring and collecting human-physiological responses during HRC Task. The main objective of this work is to identify any anomaly in the human-operator's physical or mental state during the HRC task, and using this information to change the robot behavior for better and intuitive HRC (refer Section 5.4 and also Figure 7.1(c)).

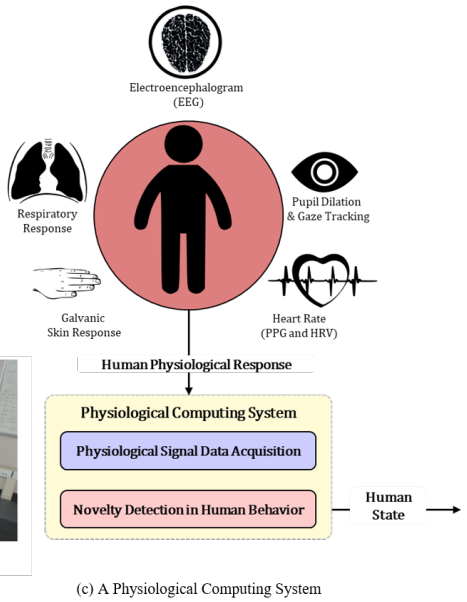
In the next section, we present the future work of this research.



(a) Human-Position Tracking based on on-robot ToF Sensor Arrays



(b) A Cyber-Physical Testbed using Digital-Twin for Task Planning and Execution



(c) A Physiological Computing System

Figure 7.1: The three impacted researches: (a) Human-Position Tracking based on on-robot ToF Sensor Arrays [8,30] (Video-Link), (b) A Cyber-Physical Testbed using Digital-Twin for Task Planning and Execution [9,31] (Video-Link), (c) A Physiological Computing System [3] (Video-Link).

7.3 Future Work

In this research, we considered the human-robot separation distance and relative human-robot velocities for the formulation of dynamic SSM using ToF sensor arrays. We also presented the consideration of the robot's payload shape and mass in the SSM formulation using effective mass and surface area vector. The future direction of this research is expanding the SSM formulation to incorporate human-robot distance, relative velocities and the mass and shape of the robot's payload, and performing human-subject experiments to measure the perceived safety of the human-operator sharing the workspace.

The results presented in this research are specific to the standard pick-place and an assembly task, which represents a human-robot coexistence scenario of HRC. The next step is investigating the performance of the proposed ToF-based dynamic safety measure in varying HRC scenarios such as hand-over tasks, hand-guiding, and shared or cooperation tasks. In the future, we intend to perform more human-subject experiments to evaluate and

analyze the effects of the ToF-based dynamic setup in an industrial setting with experienced human-operators that work with heavy-machinery and robots.

Here, an 8-node ToF sensor array, Prototype v1, was used to evaluate and validate the efficacy of using on-robot sensors for dynamic SSM. There is a scope for a lot of improvement in the design and implementation of the ToF sensor array hardware. We implemented a modular 32-node ToF sensor array, Prototype v2, to address the limitation and drawbacks of Prototype v1. The future experiments and testing will be done using the ToF-sensor array, Prototype v2. The implemented Prototype v2 can be further improved by using FPGAs that would provide faster and parallel processing of the sensor data. In the future, other on-robot sensor alternatives such as different ToF sensor chipsets and mm-wave radars will be explored for safety during HRC.

A crucial future step in this research is upgrading the ToF hardware and software in compliance with the industrial standards. The ToF sensor would be upgraded under the requirements given in IEC 61496-3:2018 [127]. This specification provides additional requirements for employing *active opto-electronic protective devices responsive to diffuse reflection* (AOPDDRs), like the ToF sensor array for safety-related systems. For compliance, this would require the inclusion of redundancy measures, error logging, and monitoring the state of the ToF sensor modules and the sensor arrays. A risk assessment as per ISO 10218 [12] and ISO 13849 safety [13, 128] for machinery standard would be performed for test setups of common HRC scenarios using the ToF-based dynamic safety measure.

The future objective of this research is to provide a Time-of-Flight-based dynamic SSM safety solution that is compliant to the industrial standards and useful as an electronic safety measure for implementing safe and flexible human-robot collaboration applications in the industry.

Bibliography

[1] S. Kumar, S. Arora, and F. Sahin, "Speed and separation monitoring using on-robot time-of-flight laser-ranging sensor arrays," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1684–1691. [Online]. Available: https://doi.org/10.1109/COASE.2019.8843326

[2] S. Kumar and F. Sahin, "Sensing volume coverage of robot workspace using on-robot time-of-flight sensor arrays for safe human robot interaction," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE, 2019, pp. 378–384.

[3] C. Savur, S. Kumar, and F. Sahin, "A framework for monitoring human physiological response during human robot collaborative task," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE, 2019, pp. 385–390.

[4] O. A. Adamides, A. S. Modur, S. Kumar, and F. Sahin, "A time-of-flight on-robot proximity sensing system to achieve human detection for collaborative robots," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1230–1236.

[5] C. Savur, S. Kumar, S. Arora, T. Hazbar, and F. Sahin, "Hrc-sos: Human robot collaboration experimentation platform as system of systems," in *2019 14th Annual Conference System of Systems Engineering (SoSE)*, May 2019, pp. 206–211.

[6] S. Kumar, C. Savur, and F. Sahin, "Dynamic awareness of an industrial robotic arm using time-of-flight laser-ranging sensors," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2018, pp. 2850–2857.

[7] S. Kumar and F. Sahin, "A framework for an adaptive human-robot collaboration approach through perception-based real-time adjustments of robot behavior in industry," in *2017 12th System of Systems Engineering Conference (SoSE)*. IEEE, 2017, pp. 1–6.

[8] S. Arora, S. Kumar, and F. Sahin, "Human position detection & tracking with on-robot time-of-flight laser ranging sensors," *arXiv preprint arXiv:1909.09750*, 2019.

[9] T. Hazbar, S. Kumar, and F. Sahin, "Cyber-physical testbed for human-robot collaborative task planning and execution," *arXiv preprint arXiv:1905.00199*, 2019.

[10] M. Bjorn and ABB Corporate Research, "Industrial Safety Requirements for Collaborative Robots and Applications," *ERF 2014 "Workshop: Workspace Safety in Industrial Robotics: trends, integration and standards*, 2014.

[11] Universal Robot, "Technical Specifications UR5," p. 5260, 2016.

[12] ISO, "ISO 10218-2:2011," International Organization for Standardization, 2011. [Online]. Available: https://www.iso.org/obp/ui/{#}iso:std:iso:10218:-2:ed-1:v1:en

[13] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[14] N. Tolós Pons, "Estandarization in human robot interaction," 2013.

[15] X. V. Wang, Z. Kemény, J. Váncza, and L. Wang, "Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation," *CIRP annals*, vol. 66, no. 1, pp. 5–8, 2017.

[16] S. Thiemermann, *Direkte Mensch-Roboter-Kooperation in der Kleinteilemontage mit einem SCARA-Roboter*. OPUS - Publication Server of the University of Stuttgart, 2005.

[17] A. BAUER, D. WOLLHERR, and M. BUSS, "Human–Robot Collaboration: a Survey," *International Journal of Humanoid Robotics*, vol. 05, no. 01, pp. 47–66, 2008. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0219843608001303

[18] P. A. Lasota, T. Fong, J. A. Shah *et al.*, "A survey of methods for safe human-robot interaction," *Foundations and Trends® in Robotics*, vol. 5, no. 4, pp. 261–349, 2017.

[19] G. B. Avanzini, N. M. Ceriani, A. M. Zanchettin, P. Rocco, and L. Bascetta, "Safety control of industrial robots based on a distributed distance sensor," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2127–2140, 2014.

[20] N. M. Ceriani, G. B. Avanzini, A. M. Zanchettin, L. Bascetta, and P. Rocco, "Optimal placement of spots in distributed proximity sensors for safe human-robot interaction," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 5858–5863.

[21] G. Hoffman, "Evaluating fluency in human–robot collaboration," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 3, pp. 209–218, 2019.

[22] J. Rasmussen, "Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models," *IEEE transactions on systems, man, and cybernetics*, no. 3, pp. 257–266, 1983.

[23] Blog, "Choosing the right distance sensor for your application," 2019. [Online]. Available: https://www.terabee.com/choosing-right-distance-sensor-your-application/

[24] iPi Motion capture, "iPi Motion capture software documentation," iPiSoft. [Online]. Available: http://docs.ipisoft.com/User_Guide_for_Multiple_Depth_Sensors_Configuration#Kinect_2_for_Windows_and_Kinect_for_Xbox_One

[25] "Freemove." [Online]. Available: https://www.veobot.com/freemove

[26] Vl53l0x - world smallest time-of-flight (tof) ranging sensor. STMicroelectronics. [Online]. Available: https://www.st.com/content/st_com/en/products/imaging-and-photonics-solutions/proximity-sensors/vl53l0x.html

[27] Vl53l1x - long distance ranging time-of-flight sensor based on st flightsense technology. STMicroelectronics. [Online]. Available: https://www.st.com/en/imaging-and-photonics-solutions/vl53l1x.html

[28] O. A. Adamides, "A time of flight on-robot proximity sensing system for collaborative robotics," 2019. [Online]. Available: https://scholarworks.rit.edu/theses/10053

[29] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 1321–1326.

[30] S. Arora, "Perception methods for speed and separation monitoring using time-of-flight sensor arrays," 2019. [Online]. Available: https://scholarworks.rit.edu/theses/10334

[31] T. Hazbar, "Task planning and execution for human robot team performing a shared task in a shared workspace," 2019. [Online]. Available: https://scholarworks.rit.edu/theses/10198/

[32] J. Bruner and B. Kisgergely, "THE DIGITAL FACTORY REPORT Digital Transformation in Manufacturing," Tech. Rep.

[33] B. Matthias, "Iso / ts 15066 - collaborative robots present status," 2016. [Online]. Available: https://www.researchgate.net/publication/282809861_ISOTS_15066_-_Collaborative_Robots_-_Present_Status

[34] GEReports, "A shortage of skilled workers threatens manufacturing rebound," Republic 3.0, 2016. [Online]. Available: http://www.gereports.com/post/115317859023/a-shortage-of-skilled-workers-threatens-manufacturings-r/

[35] B. of Labor Staistics, "Employment, hours, and earnings from the current employment statistics survey (national)," United States Labor Depatment, 2016, cES3000000001. [Online]. Available: http://data.bls.gov/timeseries/CES3000000001?data_tool=XGtable

[36] L. Bainbridge, "Ironies of automation," in *Analysis, design and evaluation of man–machine systems*. Elsevier, 1983, pp. 129–135.

[37] B. Marr, "The 4 ds of robotization: Dull, dirty, dangerous and dear," 2017. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2017/10/16/the-4-ds-of-robotization-dull-dirty-dangerous-and-dear/#6705334a3e0d

[38] NSF, "Information and intelligent systems: Advancing human-centered computing, information integration and informatics, and robust intelligence," (Archived), National Science Foundation, May 2006. [Online]. Available: http://www.nsf.gov/pubs/2006/nsf06572/nsf06572.pdf

[39] NSF, "Information and intelligent systems: Cyber-human systems (chs), information integration and informatics (iii), and robust intelligence (ri),"

(Archived), National Science Foundation, October 2016. [Online]. Available: https://www.nsf.gov/pubs/2016/nsf16581/nsf16581.pdf

[40] W. K. H. Ko, Y. Wu, K. P. Tee, and J. Buchli, "Towards Industrial Robot Learning from Demonstration," *Proceedings of the 3rd International Conference on Human-Agent Interaction*, no. October, pp. 235–238, 2015. [Online]. Available: http://doi.acm.org/10.1145/2814940.2814984

[41] C. Thomas, B. Matthias, and B. Kuhlenkötter, "Human - Robot Collaboration – New Applications in Industrial Robotics," *COMA 16 International Conference on Competitive Manufacturing Human-Robot*, no. June, 2016.

[42] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Human factors*, vol. 46, no. 1, pp. 50–80, 2004.

[43] ISO, "ISO/TS 15066:2016 - robots and robotic devices – collaborative robots." [Online]. Available: http://www.iso.org/

[44] P. Rybski, P. Anderson-Sprecher, D. Huber, C. Niessl, and R. Simmons, "Sensor fusion for human safety in industrial workcells," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 3612–3619.

[45] B. Siciliano and O. Khatib, *Springer handbook of robotics*. Springer, 2016.

[46] J. A. Marvel and R. Norcross, "Implementing speed and separation monitoring in collaborative robot workcells," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 144–155, Apr. 2017.

[47] P. Svarny, Z. Straka, and M. Hoffmann, "Toward safe separation distance monitoring from RGB-D sensors in human-robot interaction," *arXiv:1810.04953 [cs]*, Oct. 2018, arXiv: 1810.04953. [Online]. Available: http://arxiv.org/abs/1810.04953

[48] M. Zlatanski, P. Sommer, F. Zurfluh, S. G. Zadeh, A. Faraone, and N. Perera, "Machine perception platform for safe human-robot collaboration," in *2019 IEEE SENSORS*, Oct 2019, pp. 1–4.

[49] P. Long, C. Chevallereau, D. Chablat, and A. Girin, "An industrial security system for human-robot coexistence," *Industrial Robot: An International Journal*, 2018.

[50] M. Geravand, F. Flacco, and A. De Luca, "Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture," *IEEE International Conference on Robotics and Automation*, pp. 4000–4007, 2013. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6631141

[51] S. Haddadin, A. D. Luca, and A. Albu-Schäffer, "Robot Collisions: A Survey on Detection, Isolation, and Identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, Dec. 2017.

[52] S. MacPhedran. (2018) Augmented manufacturing: The big six hololens use cases for manufacturers. [Online]. Available: https://blog.smith.co/2018/augmented-manufacturing

[53] E. Matsas, G.-C. Vosniakos, and D. Batras, "Prototyping proactive and adaptive techniques for human-robot collaboration in manufacturing using virtual reality," *Robotics and Computer-Integrated Manufacturing*, vol. 50, pp. 168–180, 2018.

[54] M. Schluse, L. Atorf, and J. Rossmann, "Experimentable digital twins for model-based systems engineering and simulation-based development," in *2017 Annual IEEE International Systems Conference (SysCon)*, Apr. 2017, pp. 1–8.

[55] M. Safeea and P. Neto, "Minimum distance calculation using laser scanner and IMUs for safe human-robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 58, pp. 33–42, Aug. 2019.

[56] L. Peternel, N. Tsagarakis, D. Caldwell, and A. Ajoudani, "Robot adaptation to human physical fatigue in human–robot co-manipulation," *Autonomous Robots*, vol. 42, no. 5, pp. 1011–1021, 2018.

[57] D. Kulíc and E. Croft, "Affective state estimation for human-robot interaction," in *IEEE Transactions on Robotics*, vol. 23, no. 5, 2007, pp. 991–1000. [Online]. Available: http://ieeexplore.ieee.org/document/4339537/

[58] P. Bonifacci, L. Desideri, and C. Ottaviani, "Familiarity of faces: Sense or feeling? An exploratory investigation with eye movements and skin conductance," *Journal of Psychophysiology*, vol. 29, no. 1, pp. 20–25, 2015.

[59] "Universal Robot - Safety Functions." [Online]. Available: https://www.universal-robots.com/media/1801971/ur-g3-safety-functions-20180418.pdf

[60] F. Flacco, T. Kröger, A. D. Luca, and O. Khatib, "A Depth Space Approach for Evaluating Distance to Objects - with Application to Human-Robot Collision Avoidance," *Journal of Intelligent and Robotic Systems*, vol. 80, pp. 7–22, 2015.

[61] DAQRI, "Depth cameras for mobile ar: From iphones to wearables and beyond," 2018. [Online]. Available: https://medium.com/\spacefactor\@m{}DAQRI/depth-cameras-for-mobile-ar-from-iphones-to-wearables-and-beyond-ea29758ec280

[62] S. Yun, J. Choi, and C. Won, "Omnidirectional 3d point clouds using dual kinect sensors," *Journal of Sensors*, vol. 2019, pp. 1–17, 01 2019.

[63] C. Morato, K. Kaipa, B. Zhao, and S. K. Gupta, "Safe human robot interaction by using exteroceptive sensing based human modeling," in *Volume 2A: 33rd Computers and Information in Engineering Conference*, 2013, p. V02AT02A073.

[64] C. Morato, "Computational foundations for safe and efficient human-robot collaboration in assembly cells," Ph.D. dissertation, University of Maryland, 2016.

[65] H. S. Koppula, A. Jain, and A. Saxena, "Anticipatory planning for human-robot teams," in *Springer Tracts in Advanced Robotics*, 2016, vol. 109, no. i, pp. 453–470.

[66] P. A. Lasota and J. A. Shah, "Analyzing the effects of human-aware motion planning on close-proximity human–robot collaboration," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 57, no. 1, pp. 21–33, 2015.

[67] S. Nikolaidis, E. Kasneci, and S. Srinivasa, "Leveraging Eye Tracking and Physiological Signals for Fluent Human Robot Collaboration."

[68] "Flex 13 - An affordable motion capture camera." [Online]. Available: http://optitrack.com/products/flex-13/index.html

[69] A. Cirillo, F. Ficuciello, C. Natale, S. Pirozzi, and L. Villani, "A Conformable Force/Tactile Skin for Physical Human-Robot Interaction," *Robotics and Automation Letters, IEEE*, vol. 1, no. 1, pp. 41–48, 2016.

[70] T. Schlegl, T. Kröger, A. Gaschler, O. Khatib, and H. Zangl, "Virtual whiskers — Highly responsive robot collision avoidance," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 5373–5379.

[71] B. Lacevic and P. Rocco, "Kinetostatic danger field - a novel safety assessment for human-robot interaction," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2010, pp. 2169–2174.

[72] G. Evans, J. Miller, M. I. Pena, A. MacAllister, and E. Winer, "Evaluating the microsoft hololens through an augmented reality assembly application," in *Degraded Environments: Sensing, Processing, and Display 2017*, vol. 10197. International Society for Optics and Photonics, 2017, p. 101970V.

[73] H. Eschen, T. Kötter, R. Rodeck, M. Harnisch, and T. Schüppstuhl, "Augmented and virtual reality for inspection and maintenance processes in the aviation industry," *Procedia manufacturing*, vol. 19, pp. 156–163, 2018.

[74] A. Y. Nee, S. Ong, G. Chryssolouris, and D. Mourtzis, "Augmented reality applications in design and manufacturing," *CIRP annals*, vol. 61, no. 2, pp. 657–679, 2012.

[75] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," http://pybullet.org, 2016–2018.

[76] M. Freese, "The robot simulator coppeliasim (from the creators of v-rep)," 2019. [Online]. Available: http://www.coppeliarobotics.com

[77] T. Cichon and J. Rossmann, "Simulation-based user interfaces for digital twins: Pre-, in-, or post-operational analysis and exploration of virtual testbeds," *31st Annual European Simulation and Modelling Conference 2017, ESM 2017*, pp. 365–372, 2017.

[78] M. Laffranchi, N. Tsagarakis, and D. Caldwell, "Safe human robot interaction via energy regulation control," 11 2009, pp. 35 – 41.

[79] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "On making robots understand safety: Embedding injury knowledge into control," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1578–1602, 2012.

[80] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, Feb. 1987.

[81] T. Kröger, "Opening the door to new sensor-based robot applications—The Reflexxes Motion Libraries," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 1–4.

[82] C.-S. Tsai, *Online Trajectory Generation for Robot Manipulators in Dynamic Environment : An Optimization-based Approach*. University of California, Berkeley, 2014.

[83] C. Vogel, C. Walter, and N. Elkmann, "A projection-based sensor system for safe physical human-robot collaboration," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5359–5364.

[84] M. Safeea, N. Mendes, and P. Neto, "Minimum Distance Calculation for Safe Human Robot Interaction," *Procedia Manufacturing*, vol. 11, pp. 99–106, Jan. 2017.

[85] J. A. Marvel, "Performance Metrics of Speed and Separation Monitoring in Shared Workspaces," *IEEE Transactions on Automation Science and Engineering*, vol. 10, pp. 405–414, 2013.

[86] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Safety evaluation of physical human-robot interaction via crash-testing." in *Robotics: Science and Systems*, vol. 3, 2007, pp. 217–224.

[87] X. V. Wang, Z. Kemény, J. Váncza, and L. Wang, "Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation," *CIRP annals*, vol. 66, no. 1, pp. 5–8, 2017.

[88] V. Riley, "A general model of mixed-initiative human-machine systems," in *Proceedings of the Human Factors Society Annual Meeting*, vol. 33, no. 2. SAGE Publications Sage CA: Los Angeles, CA, 1989, pp. 124–128.

[89] G. Hoffman and C. Breazeal, "Collaboration in Human-Robot Teams," *AIAA 1st Intelligent Systems Technical Conference*, pp. 1–18, 2004. [Online]. Available: http://arc.aiaa.org/doi/10.2514/6.2004-6434

[90] W. Wei, Q. Jia, Y. Feng, and G. Chen, "Emotion Recognition Based on Weighted Fusion Strategy of Multichannel Physiological Signals," *Computational Intelligence and Neuroscience*, vol. 2018, no. 1, pp. 1–9, 2018. [Online]. Available: https://www.hindawi.com/journals/cin/2018/5296523/

[91] S. H. Fairclough, "Fundamentals of physiological computing," *Interacting with Computers*, vol. 21, no. 1-2, pp. 133–145, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.intcom.2008.10.011

[92] ——, *Advances in Physiological Computing*, 2014. [Online]. Available: http://link.springer.com/10.1007/978-1-4471-6392-3

[93] L. Tiberio, A. Cesta, and M. Belardinelli, "Psychophysiological Methods to Evaluate User's Response in Human Robot Interaction: A Review and Feasibility Study," *Robotics*, vol. 2, no. 2, pp. 92–121, 2013. [Online]. Available: http://www.mdpi.com/2218-6581/2/2/92/

[94] M. Ali, F. Al Machot, A. H. Mosa, M. Jdeed, E. Al Machot, and K. Kyamakya, "A globally generalized emotion recognition system involving different physiological signals," *Sensors (Switzerland)*, vol. 18, no. 6, pp. 1–20, 2018.

[95] D. Kulić and E. Croft, "Physiological and subjective responses to articulated robot motion," *Robotica*, vol. 25, no. 1, pp. 13–27, 2007.

[96] K. Gouizi, F. Bereksi Reguig, and C. Maaoui, "Emotion recognition from physiological signals," *Journal of Medical Engineering and Technology*, vol. 35, no. 6-7, pp. 300–307, 2011.

[97] C. Savur and F. Sahin, "American Sign Language Recognition system by using surface EMG signal," *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings*, pp. 2872–2877, 2017.

[98] S. Rohrmann, J. Hennig, and P. Netter, "Changing psychobiological stress reactions by manipulating cognitive processes," *International Journal of Psychophysiology*, vol. 33, no. 2, pp. 149–161, 1999.

[99] G. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, and J. Peters, "A probabilistic framework for semi-autonomous robots based on interaction primitives with phase estimation," in *Robotics Research*. Springer, 2018, pp. 253–268.

[100] G. Tang, P. Webb, and J. Thrower, "The development and evaluation of robot light skin: A novel robot signalling system to improve communication in industrial human–robot collaboration," *Robotics and Computer-Integrated Manufacturing*, vol. 56, pp. 85–94, 2019.

[101] J.-P. Couderc, S. Kyal, L. K. Mestha, B. Xu, D. R. Peterson, X. Xia, and B. Hall, "Detection of atrial fibrillation using contactless facial video monitoring," *Heart Rhythm*, vol. 12, no. 1, pp. 195–201, 2015.

[102] G. R. Tsouri and Z. Li, "On the benefits of alternative color spaces for noncontact heart rate measurements using standard red-green-blue cameras," *Journal of biomedical optics*, vol. 20, no. 4, p. 048002, 2015.

[103] J. H. Guzman, J.-P. Couderc, and G. R. Tsouri, "Accurate hemodynamic sensing using video plethysmography with high quality cameras," in *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*. IEEE, 2019, pp. 1–6.

[104] A. Casalino, C. Messeri, M. Pozzi, A. M. Zanchettin, P. Rocco, and D. Prattichizzo, "Operator awareness in human–robot collaboration through wearable vibrotactile feedback," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4289–4296, 2018.

[105] "Vr training and enterprise applications need haptic gloves," Jul 2019. [Online]. Available: https://haptx.com/solutions/

[106] ZeroMQ, "Zeromq (Ømq, 0mq or zmq) -a high-performance asynchronous messaging library," 2020. [Online]. Available: https://zeromq.org/

[107] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[108] SCCN, "Lab Stream Layer (LSL)," 2018.

[109] G. Hoffman and C. Breazeal, "Effects of anticipatory action on human-robot teamwork," *International Conference on Human-Robot Interaction*, pp. 1–8, 2007.

[110] D. Kulic and E. Croft, "Anxiety detection during human-robot interaction," *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 389–394, 2005.

[111] N. Pedrocchi, F. Vicentini, M. Matteo, and L. M. Tosatti, "Safe Human-Robot Cooperation in an Industrial Environment," *International Journal of Advanced Robotic Systems*, vol. 10, no. 1, p. 27, Jan. 2013. [Online]. Available: https://doi.org/10.5772/53939

[112] T. Instruments. mmwave sensors: Intelligent autonomy at the edge with single-chip millimeter-wave sensors. [Online]. Available: http://www.ti.com/sensors/mmwave/overview.html

[113] K. Dröder, P. Bobka, T. Germann, F. Gabriel, and F. Dietrich, "A Machine Learning-Enhanced Digital Twin Approach for Human-Robot-Collaboration," *Procedia CIRP*, vol. 76, pp. 187–192, Jan. 2018.

[114] E. Negri, L. Fumagalli, and M. Macchi, "A review of the roles of digital twin in cps-based production systems," *Procedia Manufacturing*, vol. 11, pp. 939–948, 2017.

[115] A few words about spad (single photon avalanche diodes) technology. POLIS-Pilot Optical line for imaging and sensing. [Online]. Available: http://polis.minalogic.net/SPAD_TOF.php

[116] J. Kvam, "Time-of-flight principles, challenges, and performance," 2017. [Online]. Available: https://www.st.com/content/dam/technology-tour-2017/session-1_track-4_time-of-flight-technology.pdf

[117] Mappydot - a smarter micro lidar sensor. SensorDots. [Online]. Available: https://sensordots.org/

[118] M. Kok, J. D. Hol, and T. B. Schön, "Using inertial sensors for position and orientation estimation," *arXiv preprint arXiv:1704.06053*, 2017.

[119] Y. Lin, H. Min, and H. Wei, "Inertial measurement unit–based iterative pose compensation algorithm for low-cost modular manipulator," *Advances in Mechanical Engineering*, vol. 8, no. 1, p. 1687814015626850, 2016.

[120] J. Heinzmann and A. Zelinsky, "Quantitative safety guarantees for physical human-robot interaction," *The International Journal of Robotics Research*, vol. 22, no. 7-8, pp. 479–504, 2003.

[121] S. Gottschalk, M. C. Lin, and D. Manocha, "Obbtree: A hierarchical structure for rapid interference detection," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 171–180. [Online]. Available: https://doi.org/10.1145/237170.237244

[122] S. Maneewongvatana and D. M. Mount, "On the efficiency of nearest neighbor searching with data clustered in lower dimensions," in *Computational Science — ICCS 2001*, V. N. Alexandrov, J. J. Dongarra, B. A. Juliano, R. S. Renner, and C. J. K. Tan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 842–851.

[123] O. Ogorodnikova, "How safe the human-robot coexistence is ? theoretical presentation," 2009. [Online]. Available: http://uni-obuda.hu/journal/Ogorodnikova_20.pdf

[124] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework," *The international journal of robotics research*, vol. 14, no. 1, pp. 19–36, 1995.

[125] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 4193–4198.

[126] A. B. Downey, *Think Stats*. O'Reilly Media, Inc., 2011.

[127] IEC, "IEC 61496-3:2018 - safety of machinery - electro-sensitive protective equipment - part 3: Particular requirements for active opto-electronic protective devices responsive to diffuse reflection (aopddr)." [Online]. Available: https://webstore.ansi.org/standards/iec/iec61496ed2018

[128] ISO, "ISO13849:2015 - safety of machinery." [Online]. Available: http://www.iso.org/

[129] R. Szeliski, "Rapid octree construction from image sequences," *CVGIP: Image understanding*, vol. 58, no. 1, pp. 23–32, 1993.

[130] L. Shao and H. Zhou, "Curve fitting with bezier cubics," *Graphical models and image processing*, vol. 58, no. 3, pp. 223–232, 1996.

[131] T. Huckaby, "Curved axis revolutions," 05 2013. [Online]. Available: https://www.researchgate.net/publication/264309866_Curved_Axis_Revolutions

[132] A. W. Goodman and G. Goodman, "Generalizations of the theorems of pappus," *The American Mathematical Monthly*, vol. 76, no. 4, pp. 355–366, 1969. [Online]. Available: http://www.jstor.org/stable/2316426

Appendix

A. Calculating Robot Effective Mass

The effective mass is defined at an operational point on the robot moving in the direction of impact/collision with the human. The effective mass reflects the inertial properties of the robot at the point of impact. The effective mass is equivalent to the mass of a rigid body having the same kinetic energy at the operational point on the robot. The effective mass formulation and derivation have been defined in detail in [123, 124].

For a multi-link manipulator (n degrees-of-freedom), the effective mass (m_{eff}) changes with the robot configuration ($\boldsymbol{q} = [q_1, q_2, \ldots, q_n]^T$). The danger posed by an impact/collision is only considered in the operational space (also known as the task space, where $\boldsymbol{x} = \left[p_x, p_y, p_z, \theta_x, \theta_y, \theta_z\right]^T$ is the position and orientation of the end-effector), and not in the joint space of the robot. Thus, the effective mass is defined in the operational space of the robot. The inertial/mass properties of the robot are generally expressed with respect to its motion in the joint space.

The equations of motion (dynamic model) of a robot manipulator in joint ($\boldsymbol{q}^{n\times1}$) and operational ($\boldsymbol{x}^{6\times1}$) spaces is described in [124] and are defined as

$$\boldsymbol{M(q)\ddot{q} + b(q,\dot{q}) + g(q) = \tau}$$
$$\boldsymbol{\Lambda(q)\ddot{x} + \mu(q,\dot{q}) + p(q) = F}$$

where $\boldsymbol{M(q)}^{n\times n}$ is the inertial/mass matrix and the $\boldsymbol{\Lambda(q)}^{6\times6}$ is the robot end-effector's 'pseudo kinetic energy' matrix. The $\boldsymbol{b(q,\dot{q})}^{n\times1}$, $\boldsymbol{\mu(q,\dot{q})}^{6\times1}$ are the vectors of centrifugal and Coriolis forces in the robot's joint and operational space respectively, and $\boldsymbol{g(q)}^{n\times1}$, $\boldsymbol{p(q)}^{6\times1}$ are the vectors for gravity. The vectors $\boldsymbol{\tau}^{n\times1}$ and $\boldsymbol{F}^{6\times1}$ are the generalized vectors of joint torques and the end-effector force respectively.

The relation between the inertial/mass matrix $\boldsymbol{M(q)}$ and $\boldsymbol{\Lambda(q)}$ has been defined in [124] as

$$\boldsymbol{\Lambda(q) = \left(J(q)M^{-1}(q)J^T(q)\right)^{-1}}$$

where $\boldsymbol{J(q)}^{6\times n}$ is the basic Jacobian associated with the n-DOF robot end-effector's linear($\boldsymbol{v}$) and angular($\boldsymbol{\omega}$) velocities and $\boldsymbol{M(q)}$ is a symmetric positive defined inertia/mass matrix of

the robot. It is assumed that the impact/collision is transient (for a very short time and close distance) occurs as a result of the robot's transitional motion. Hence, the Jacobian associated with only the linear velocity, $\boldsymbol{J_v(q)}^{3\times n}$ is considered for calculation of the 'pseudo kinetic energy' ($\boldsymbol{\Lambda_v(q)}^{3\times 3}$) matrix. The effective mass scalar m_{eff} in the direction of impact $\hat{\boldsymbol{u}}^{3\times 1}$ can be defined as

$$\begin{aligned} \boldsymbol{\Lambda_v^{-1}(q)} &= J_v(q)M^{-1}(q)J_v^T(q) \\ \frac{1}{m_{eff}(\boldsymbol{\Lambda_v(q)})} &= \hat{\boldsymbol{u}}^T\Lambda_v^{-1}(q)\hat{\boldsymbol{u}} \end{aligned}$$

For evaluation of the effective mass in a given direction, the matrix $\boldsymbol{\Lambda_v(q)}$ is diagonalized to avoid coupling of the elements. The eigenvalues ($\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \lambda_3\}$) of $\boldsymbol{\Lambda_v(q)}$ can be used to represent an ellipsoid geometrical approximation of the effective-mass at the operational point with elliptic radii $\{\sqrt{\lambda_1}, \sqrt{\lambda_2}, \sqrt{\lambda_3}\}$ [123, 124] as

$$\left(\frac{x}{\sqrt{\lambda_1}}\right)^2 + \left(\frac{y}{\sqrt{\lambda_2}}\right)^2 + \left(\frac{z}{\sqrt{\lambda_3}}\right)^2 = 1$$

These eigenvalues associated with $\boldsymbol{\Lambda_v(q)}$ provide useful characterization of the bounds of the magnitude of the effective mass ($m_{min} = \min\{\boldsymbol{\lambda}\}, m_{max} = \max\{\boldsymbol{\lambda}\}$).

B. Event-Marker Generation during HRC Experiments

An important feature of the cyber-physical system that is implemented, is event marker generation during the experiment. In the experiment, important events need to be investigated and generated during the experiment. The event markers help synchronize signal across different channels. For example, extracting physiological signals during signal recording between event-marks representing the *"Experiment Start"* and *"Experiment End"*, or defining an HRC state when the *"Robot is approaching"* the human become trivial problems. Thus, the markers can be used during post-processing for efficient data segmentation and epoching. For the SSM safety configuration setup, event markers based on human position in safety zones and human task progress are generated. A list of events that are automatically generated during the HRC task for the experiments in this research are listed in Table 1 below:

Table 1: The table shows the event-markers used during the experiment

	Event Marker	Definition
Task-based Events	Experiment Start	the experiment has started
	Robot Task-Start	robot task started.
	Robot Task-Stop	robot task stopped.
	Human Task [n]	[n]th task being performed by the human.
	Robot Task[n]	robot is currently picking [n]th part
	Robot approaching	Each time robot comes toward the human, generate an event
	Robot Pick up	robot has picked up the part
	Robot Placed	robot has placed the part
	Experiment Stop	the experiment is complete
Safety-based Events	Experiment Start	the experiment has started
	Robot state change	When robot change state between Normal, Reduced, and Stop
	Robot is stopping	When robot going to complete stop
	Robot is speeding up	When robot is going to normal speed
	Robot is slowing down	When robot is slowing down.
	Human Position in Safety Zones	based on the distance from the robot - Safe, Warning, Danger
	Experiment Stop	experiment is complete
Manual Events	Take a Look	an event manually generated by the investigator using a keyboard.
	Human Collision	a manual event where human while performing the task runs into a stopped robot.
	Collision	robot did not stop in time and resulted in collision.
	Robot Lag	robot starts to lag behind in completion of task to assist the human.
	Human Emotion	investigator notices change in human behavior.

C. More Figures from the Dynamic SSM implementation

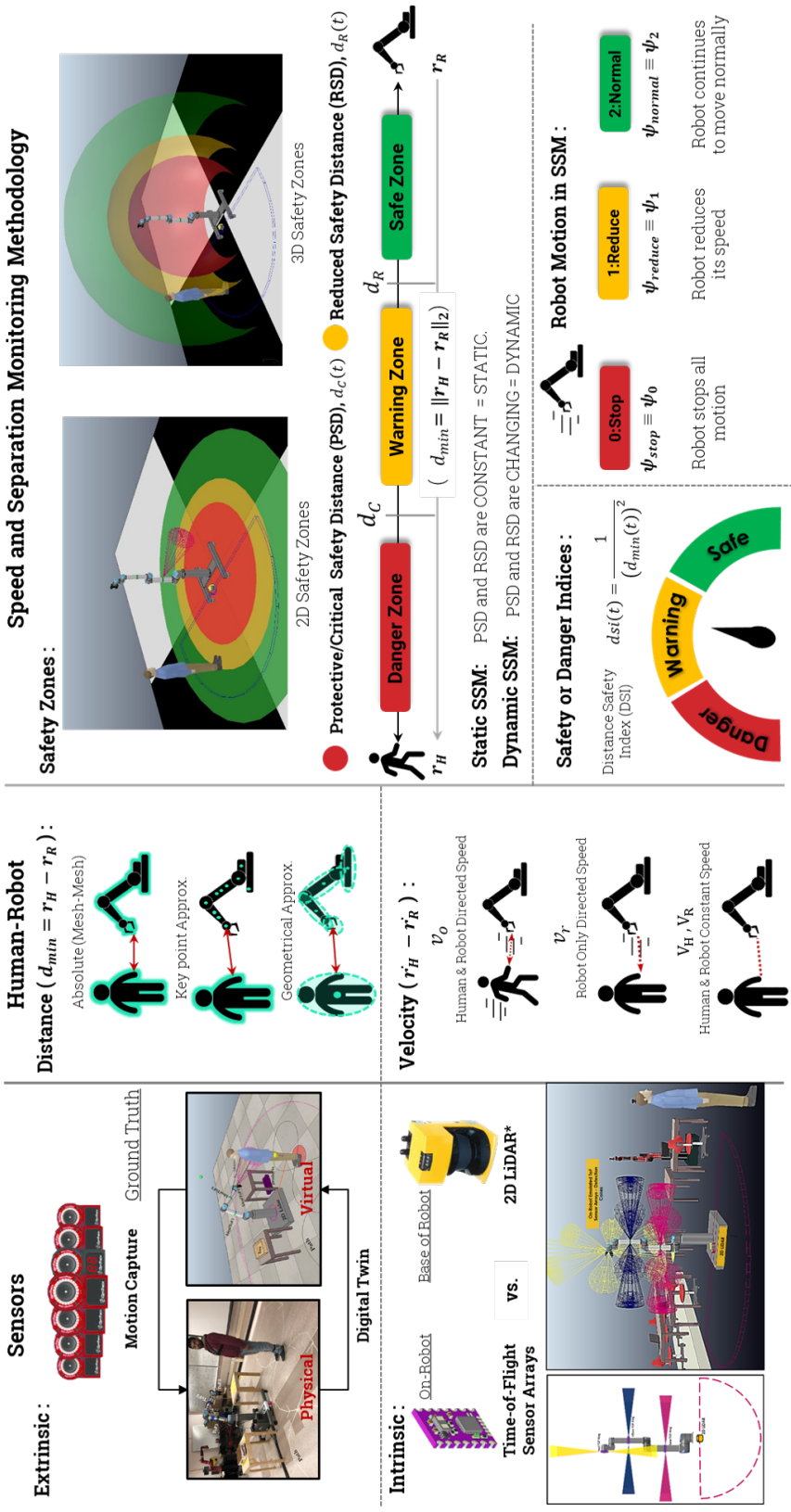


Figure C.2: An overview of the research road-map.

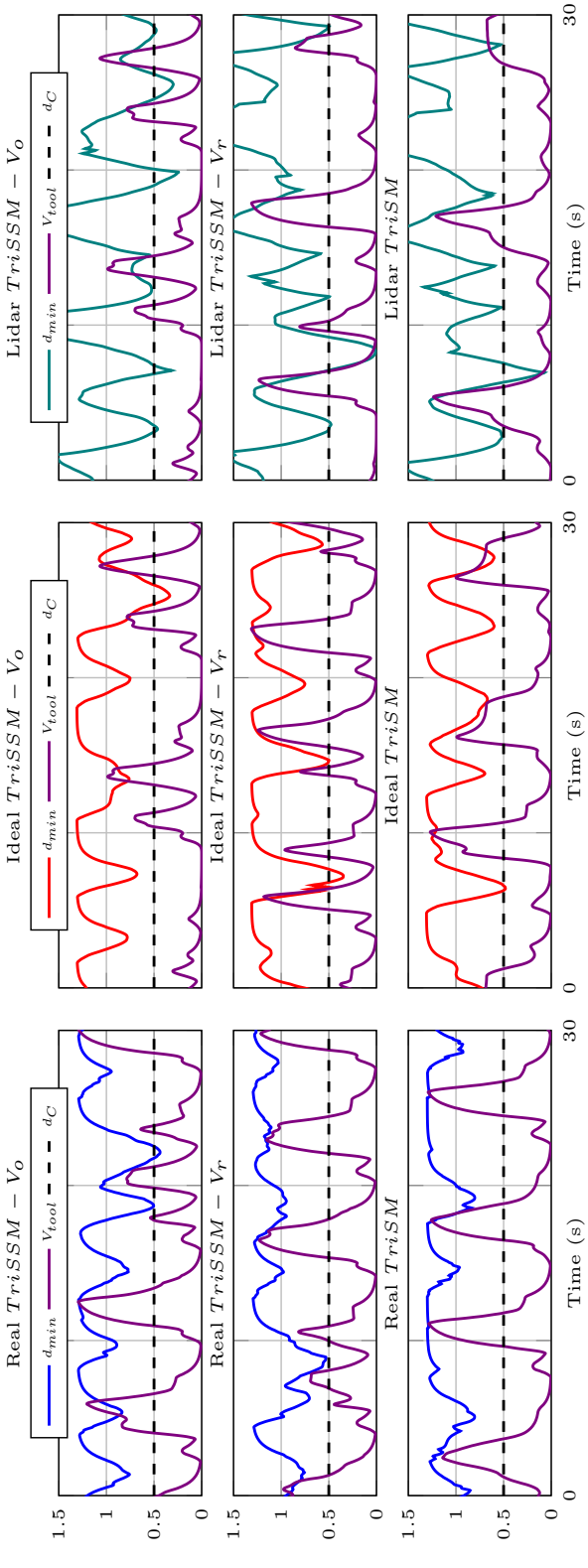


Figure C.3: Response of the robot TCP velocity ($v_{tool} \equiv v_{TCP}$) to the change in the measured minimum distance. The columns represent the minimum distance measured using - ToF sensor arrays i.e *Real*, in V-REP i.e. *Ideal* distance and 2D *Lidar*, and the rows are the SSM safety configurations - *TriSSM-Vo, TriSSM-Vr and TriSM*.

D. Human Subject Protection Plan and Research Statement

The proposed human subject research has been approved by the Rochester Institute of Technology Institutional Review Board (IRB), Human Subject Research Office (HSRO # 01052019, FWA # 00000731). This research has been categorized under 'no greater than minimal risk.' At the time of approval, the principal investigators for this research were Dr.Ferat Sahin and his Ph.D. student Shitj Kumar.

In this study, we are using Collaborative robots – Universal Robot UR10 to perform tasks while sharing the workspace with a human operator. Collaborative Robots by design are made to work with humans and have safety procedures to reduce human injury significantly. In this research, we have designed sensors to monitor robot workspace that detect the human or objects moving in its workspace and adjust its movements to ensure safety. We wish to measure the effects of the change in robot behavior in terms of Safety, Performance, and Production. Besides, we also want to measure the human physiological state, such as stress levels and attention during the experiments. The human subject sharing of the robot workspace is to perform an example task and will wear noninvasive biometric devices to record various biological signals. They will also be wearing gaze tracking glasses to determine their attention in the workspace. Along with this, the human will be given a controller to command (stop and reduce speed) button for motion control of the robot. The robot will operate at speeds as described by the ISO and RIA standards for safe human-robot interaction.

There are three main objectives for this study. Firstly, this research will help in validating the performance of the Time-of-Flight (ToF) based proximity sensors developed in the CMCR lab at RIT for its use as safety devices in a human-robot interaction setup. Secondly, the changes in robot speeds and behavior can be used as stimuli to understand and quantify human operator's comfort levels in the shared workspace i.e., the human trust in automation. And lastly, we wish to see that can the physiological signals be used to quantify trust effectively and whether this could be used to modify robot behavior based on the physiological state of the human for a more comfortable and efficient human-robot interaction. This research hopes to get insight on how the robot behavior can be modified to help the human operators in industry working with them.

As there are three main challenges to a Human-Robot Collaboration setup: Safety,

Human Trust in Automation and Productivity. Our objective is to build human trust in automation while working with the robot, ensuring human safety while optimizing the productivity of the task. In our opinion, to build human trust in automation, safety is the first step. Even though collaborative robots are inherently safe to work with, any unexpected human-robot physical contact can cause the human to be apprehensive of the robot movements and thereby have less trust in the automation. Lack of trust would, in turn, affect productivity. Thus we wish to observe and analyze the effect of increasing safety using our on-robot ToF based proximity sensors and its impact on human physiological state and productivity.

There will be two categories of subjects: healthy subjects that have experience working with machinery and subjects that do not. The common factor is that they are not mentally disabled, have normal or corrected vision, and are 18 years and older. The rational of inclusion or exclusion of any sub-population is as follows:

All subjects that are to be included in this research should have no known mental handicap, no severe physical handicap that would obstruct the subjects' motion in the robot's workspace, and normal or corrected vision. This is because, during the experiment, we wish the human subject to move around in the workspace while being aware of the robot motion. One of the main objectives of this research is to understand the human response to robot safety measures and determine how comfortable the subject is during an HRC task. Therefore, It is essential for the human subject to be able to adjust to the robot motion if necessary as well as move along a predefined path while performing the human-robot collaborative task.

The experiments will have volunteers from RIT, surrounding community, and partnered companies in Rochester. The subjects will only be identified using a unique code/experiment number. The information for the data will contain the subject's age, left/right-handed, experienced with machinery or not. Any identifiable information (such as email/name used during correspondence) will be kept on a secure MABL RIT research lab server, accessible solely to the investigators. This information will be strictly for record-keeping purposes and will not be a part of this research. Videos and Photographs will be taken for post-analysis for the investigators' reference during the experiment. The experiments performed will be 45-60minutes long per session for each subject. The data collected will be from the

biometric devices for recording physiological signals, haptic feedback devices, and a questionnaire. Questionnaire after the experiment asking questions on a Likert scale in terms of the performance of the robot and the human subject's sense of comfort and whether he/she felt safe working with the robot. Other data recorded would be based on the experiment using a Motion Capture tracking system to measure experiment parameters such as human-robot absolute minimum distance, robot Idle time, task completion time, and robot velocities. This data is not explicitly related to human subjects

There are no risks socially or legally or other. There are minimal psychological risks, as every subject might respond differently to sharing a workspace with the robot. Physical risks include:

- Scratch/Bruise or other non-severe health effects in case of human-robot collision and maybe lacerations/cuts if robot moving at max speed, which is highly unlikely.
- Little to no discomfort from wearing the electrode cap or other skin-electrodes, possible skin irritation if worn for a long time.
- Slight discomfort wearing eye-tracking glasses.
- Extremely low likelihood of experiencing electrical discharge from the biosignal recording experiment. (The electrical discharge is not perceptible).

There is a three-step risk mitigation done to avoid a human-robot physical collision:

1.) **ToF sensor arrays for proximity detection**: The ToF sensor arrays can detect human presence and slow or stop the robot based on the human-robot distance and human-robot relative speeds. The ToF sensors arrays have been tested and have proven to be responsive to the detection of humans in the robot workspace. [1]

2.) **Force/Torque Based protective Stop in-built in the Collaborative Robot**: The Collaborative Robots have a built-in safety protective stop that triggers if an external force is applied in the direction opposite to its task. The threshold is set to 50N. This ensures that in case of physical contact between the human subject and the robot, a 50N force will trigger the robot to stop. This setting will provide the reduction of any physical injury.

3.) **Human Operator (Principle Investigator) near Emergency Stop to brake the**

[1] In the attached supplemental documents, media, and published work, these have been tested (Folder *IRB Documents*). Please refer to the *Media* folder, and the *Data Collections Tools* and procedures document for more information.

robot.: In addition to the safety provided by the ToF sensors, an operator will continuously monitor the human-robot interaction during the experiment and can press the emergency stop in case of any failure. Moreover, visual feedback of the detection of humans in the workspace is shown using an LED indicator. The Led indicator has three colors

a.) **Red-Safety Stop**, showing the robot is stopping because the human is closer to the robot,

b.) **Yellow-Reduced Speed**, the robot is slowing down because the human has entered the reduced region of the robot, and

c.) **Green-Normal**, the robot is working at maximum programmed speeds because human is farther than the threshold distance, that is needed for the robot to stop and any injury to the human.

In order to preserve the confidentiality of the subject, the data recorded will not contain any identifiable information or markers. Any identifiable information taken from the subject will be stored on a secure lab server, which is password protected, has firewalls, and is connected to the secure RIT network. This information will not be disclosed and will be available only to the investigators of this study. The prospective participants will be provided with the consent forms and all the necessary information prior to the experiment. The procedure for the experiment will be explained in person by the investigator. Ensuring that the participant understands the aspects of the experiment and volunteers himself /herself to participate, he/she will be requested to sign the consent forms. The participant will be informed of his/her rights and responsibilities prior to the experiment in the consent form.

There are no direct benefits associated with participating in this research. Information from this study will be a step towards better understanding the use of biosignals as feedback to the robot to improve human-robot collaboration, validate the use of the new form of proximity detection, and allow new ideas for the process of automation based on safe human-robot interaction. This insights from this research could help develop a more efficient and flexible automation process with a human in the loop, where the robot's speed, precision, and power can be combined with human's dexterity, flexibility, creativity, and quick decision making.

The RIT IRBs require all research personnel who conduct human subjects research to

complete human subjects' protection training using the online CITI course (www.citiprogram.org/). Additionally, all staff conducting human subjects research must complete continuing education on a regular basis. Continuing education requirements may be met by attending an educational session approved by the IRB (e.g., IRB Essentials), a national conference that addresses human subjects' protections in research, or completion of additional CITI modules and quizzes. RIT's Institutional Review Board has duly certified all research personnel in this study for 'no greater than minimal risk' human subject research.